Solutions Linux

David Egan - Paul Zikopoulos

Bases de données sous Linux

Installer et configurer MySQL, PostgreSQL, Oracle8i, DB2...

Au sommaire

- MySQL
- PostgreSQL
- Oracle8 et Oracle8*i*
- IBM DB2
- Informix
- Sybase
- Progress
- Site e-commerce avec PHP et MySQL

Eyrolles

Bases de données
sous
Linux

BY dali200821

CHEZ LE MÊME ÉDITEUR

Du même auteur

G. Briard. – **Oracle8i sous Linux** (2ème tirage).
N°9135, 2000, 650 pages + CD-Rom.

G. Briard. – **Oracle8 pour Windows NT.**
N°9023, 1998, 350 pages + CD-Rom.

Bases de données

C. Soutou. – **Objet-relationnel avec Oracle8**.
N°9063, 1999, 480 pages.

G. Gardarin. – **Les bases de données.**
N°9060, 1999, 816 pages.

G. Gardarin. – **Internet/intranet et bases de données.**
N°9069, 1999, 264 pages.

Linux

P. Wainwright – **Apache professionnel**.
N°9150, 2000, 650 pages.

H. Holz, B. Schmitt, A. Tikart. – **Internet et intranet sous Linux**.
N°9101, 1999, 474 pages + CD-Rom.

C. Blaess. – **Programmation système en C sous Linux**.
N°9136, 2000, 960 pages.

N. Matthew, R. Stones. – **Programmation Linux.**
N°9129, 2000, 900 pages.

D. Odin. – **Programmation Linux avec GTK+.**
N°9115, 2000, 684 pages.

A. Berlat, J.-F. Bouchaudy, G. Goubet. – **Linux Administration.**
N°9147, 2000, 480 pages.

Développement Web/intranet

L. Lacroix, N. Leprince, C. Boggero, C. Lauer. – **Programmation Web avec PHP.**
N°9113, 2000, 382 pages.

A. Patzer. – **Programmation Java côté serveur. Servlets, JSP et EJB**
N°9109, 2000, 984 pages.

A. Tasso. – **Le livre de Java premier langage.**
N°9156, 2000, 312 pages.

N. McFarlane. – **JavaScript professionnel.**
N°9141, 2000, 950 pages.

J.-C. Bernadac, F. Knab. – **Construire une application XML.**
N°9081, 1999, 500 pages.

Bases de données sous Linux

David Egan Paul Zikopoulos

ÉDITIONS EYROLLES
61, Bld Saint-Germain
75240 Paris Cedex 05
www.editions-eyrolles.com

Traduction autorisée de l'ouvrage en langue anglaise intitulé :
DBA's Guide to Databases on Linux, including Oracle and MySQL,
by David Egan and Paul Zikopoulos.
Syngress Publishing Inc., ISBN : 1-928994-04-0

Traduction de l'anglais par Martine Chalmond

Le code de la propriété intellectuelle du 1er juillet 1992 interdit en effet expressément la photocopie à usage collectif sans autorisation des ayants droit. Or, cette pratique s'est généralisée notamment dans les établissements d'enseignement, provoquant une baisse brutale des achats de livres, au point que la possibilité même pour les auteurs de créer des œuvres nouvelles et de les faire éditer correctement est aujourd'hui menacée.
En application de la loi du 11 mars 1957, il est interdit de reproduire intégralement ou partiellement le présent ouvrage, sur quelque support que ce soit, sans autorisation de l'Éditeur ou du Centre Français d'exploitation du droit de Copie, 20, rue des Grands Augustins, 75006 Paris.
© 2000, Syngress Publishing Inc. All rights reserved.
© Éditions Eyrolles, 2000. Version eBook (ISBN) de l'ouvrage : 2-212-28068-8

Dépôt légal : septembre 2000
N° d’éditeur : 6159

Remerciements

Nous souhaitons remercier, pour leur gentillesse et leur aide, les personnes sans qui ce livre n'aurait pas vu le jour.

Richard Kristof, Duncan Anderson, Jennifer Gould, Robert Woodruff, Kevin Murray, Dale Leatherwood, Shelley Everett, Laurie Hedrick, Rhonda Harmon, Lisa Lavallee et Robert Sanregret de Global Knowledge, qui nous ont facilité l'accès aux meilleures formations professionnelles sur les systèmes d'information.

Ralph Troupe et l'équipe de Rt. 1 Solutions pour leurs recherches incomparables dans les domaines de la conception, du déploiement et de l'assistance, concernant les réseaux propres aux entreprises.

Karen Cross, Kim Wylie, Harry Kirchner, John Hays, Bill Richter, Kevin Votel, Briton Clark, Sarah Schaffer, Luke Kreinberg, Ellen Lafferty et Sarah MacLachlan de Publishers Group West, qui ont bien voulu partager leur immense expérience et leur expertise en marketing.

Peter Hoenigsberg, Mary Ging, Caroline Hird, Simon Beale, Julia Oldknow, Kelly Burrows, Jonathan Bunkell, Catherine Anderson, Peet Kruger, Pia Rasmussen, Denelise L'Ecluse, Rosanna Ramacciotti, Marek Lewinson, Marc Appels, Paul Chrystal, Femi Otesanya et Tracey Alcock de Harcourt International, qui nous ont encouragés dans notre vision d'ouverture au monde entier.

Une mention spéciale pour les professionnels d'Osborne. Nous sommes honorés de publier avec eux des ouvrages de renom, dans la série Global Knowledge Certification Press.

© copyright Éditions Eyrolles

Les auteurs

Tom Bascom est architecte d'entreprise chez EDS et responsable des consultations en matière des technologies d'entreprise et des échanges d'information entre EDS et ses clients. Il est chargé de la formulation de stratégies inter-entreprises et assume un rôle de conseil au niveau de la conception, du développement, de l'implémentation et de la gestion des produits et solutions appropriés. Tom a participé régulièrement à l'élaboration des produits utilisant la base de données Progress depuis 1986, ainsi qu'à Linux depuis 1994 ; il apporte fréquemment sa contribution à des forums publics tels que PEG ou le groupe de news comp.databases.progress. Tom se fait conférencier à l'occasion, et il est l'auteur d'articles techniques, ou spécifiques de l'industrie, dans le magazine Progressions.

Jamieson Becker est ingénieur, spécialiste de Linux/UNIX et Windows NT ; il est propriétaire de Ring0.com, une société de services web. Depuis ses débuts en informatique, en 1980, Jamieson est devenu un consultant accompli, partenaire d'IBM, Compaq, Carrier, Brown & Williamson, VoiceStream, Cargill et Citizens. Jamieson est spécialise des grandes applications web utilisant Linux. Il vit avec sa femme, Jennifer, et son chat, Angie, à Atlanta, en Georgie. On peut le joindre par e-mail à jamiebecker@ring0.com.

Geoff Crawford dirige une société de conseil implantée au New Jersey, Innov8 Computer Software, LLC (www.innov8cs.com). Il pratique Progress depuis 15 ans, à titre professionnel. Geoff est actif au sein du Progress E-mail Group, auteur de Progress ODBC FAQ et intervient dans les conférences des utilisateurs Progress. Son temps libre, Geoff le partage, en une combinaison inattendue, entre la recherche de ses origines allemandes en Pennsylvanie et l'écoute de Pantera.

© copyright Éditions Eyrolles

Mark Connoly travaille sur les bases de données relationnelles depuis 1987. Il a une grande expérience de DB2, Informix et Oracle, acquise dans des fonctions aussi diverses qu'administrateur de base de données, développeur, chef de produit et spécialiste en marketing.

Brad Dietrich (B.S. Computer Science and Electrical Engineering) a travaillé aussi bien en tant que professionnel qu'au titre de la recherche dans l'espace Internet, dès son apparition. Il est considéré comme l'un des ingénieurs sur lesquels repose Software Engineers for ChannelDot Networks, à San Francisco, CA. Avant de rejoindre ChannelDot, il a fondé et dirigé DefTech Contracting (www.deftech.com), une entreprise de développement d'applications web et internet, où il a développé des applications web mondialement connues pour de nombreux clients, dont Mattel et Spyglass. Il a conduit des projets de grande ampleur, qui vont de l'e-commerce à l'intégration de systèmes existants et de bases de données qui se mesurent en terabytes. Il a participé, au sein de la communauté open source, à des développements qui vont d'applications middleware au contournement de bogues contenus dans des pilotes d'origine propriétaire. Avant de démarrer son entreprise de développement d'applications Internet, il a développé des applications web de gestion de réseau chez FORE Systems ainsi que des sites web d'éducation à distance pour l'université Carnegie Mellon.

David Egan (P.Eng., BASc Enginering, Université de Toronto, MCT, RHCE), a vécu dans différents pays et travaillé sur des micro-ordinateurs depuis les premiers débuts d'Apple et du PC d'IBM. Le premier « travail-avec-passion » (*« hobby-turned-job »*) de David fut la programmation en langage C et en assembleur Z80, pendant cinq ans. David est consultant en matière d'intégration des systèmes VMS/UNIX/NT/PC et formateur en technologie depuis 15 ans. Il écrit des livres à l'occasion mais, la plupart du temps, il se consacre aux activités de direction et de conception de séminaires sur Unix, NT et Linux pour Global Knowledge Inc. à Cary, NC : http://www.globalknowledge.com. Quand il n'est pas sur la route à prêcher les vertus de Linux et de NT, il réside près de Vancouver, BC, avec son adorable femme Deborah et leurs enfants, Vanessa et Callen. On peut leur écrire à egand@istar.ca ou egand@wwdb.org, ou sur http://home.istar.ca/ egand.

Brent Gilmore (Sun Certified Solaris Administrator et Sun Certified Network Administrator) est administrateur Oracle et développeur pour l'université d'état de San Diego. Il travaille dans l'informatique depuis 1989 et a obtenu un certificat B.S. en systèmes d'information et un certificat M.S. en gestion d'administration des affaires. Brent se sert de Linux depuis 1996 et d'Oracle depuis 1997. Il travaille actuellement au développement d'applications Internet et à la conception de bases de données utilisant Oracle8/8i, Linux, Perl, PL/SQL, OAS et Apache.

© copyright Éditions Eyrolles

Bryan Schwab (B.S. Computer Engineering, Florida Atlantic University) est ingénieur principal chez Billpoint, une entreprise spécialisée dans les portails (*eBay*), où il est responsable du développement de la logistique (*back end*) pour l'e-commerce, dont les serveurs de commandes et d'expéditions, les spécifications de sécurité, les systèmes de monitoring et l'intégration d'applications stratégiques majeures. Après plus de 10 ans d'expérience de l'ingénierie de logiciel, Bryan se spécialise dans les architectures de base de données back end sur Linux, Solaris, FreeBSD, OpenBSD et Windows NT/9X. Il a conçu et implémenté des systèmes d'information pointus et des programmes de gestion de grande envergure pour des sociétés telles que Wired Digital ou NetGravity. Il écrit dans *Wired Magazine* et *Webmonkey.*

Tim Schaefer est administrateur de base de données Informix et consultant en data warehousing. Il travaille actuellement pour Galaxy Latin Americ DirecTV, à Fort Lauderdale, Floride. Il a travaillé sur des bases des données Informix de toutes tailles, allant du système équipé du moteur standard sur machine Xenix 80286, avec seulement 16 Mo de mémoire, jusqu'à des entrepôts de données à grande échelle équipés de Informix Extended Parallel Server (XPS). Il est dans l'informatique depuis 1985, depuis le collège (université Webster de Saint-Louis), et travaille avec les produits Informix depuis 1988. A son actif, retenons sa participation à un groupe international de développeurs et d'hommes d'affaires choisis pour promouvoir les avantages d'Informix sur Linux. Il gère aussi le site InxUtil, http://www.inxutil.com, un site web dédié aux outils Informix, et s'implique activement dans le groupe des utilisateurs Informix Sud Floride à Ft Lauderdale. Tim est aussi un adepte de la plongée sous-marine, et le mieux, pour le rencontrer, est d'aller faire un tour, certains week-ends de printemps et d'été, sur la côte de Ft. Lauderdale ; on le trouvera au milieu des poissons, des épaves et des récifs. On peut aussi joindre Tim à tschaefe@bellsouth.net

Alavoor Vasudevan « Al Dev » (B.E. Osmania University, M.S University of Texas à El Paso), a démarré sa carrière informatique en 1985. Depuis lors, il a beaucoup travaillé avec les bases de données sous Unix/C/C++/SQL. Il a une expérience du e-commerce, de Linux et de HTML. Parmi ses travaux annexes figurent l'écriture d'un bon nombre de documentations pour Linux/UNIX ainsi que le développement de logiciels pour Linux/UNIX. Al Dev vit à Pearland, Texas ; la semaine, il est consultant pour des sociétés implantées à Houston et dans les environs.

Paul Zikopoulos est un membre senior de l'équipe de développement et d'information sur DB2, spécialisé dans l'installation de DB2 et les problèmes de configuration. Il a plus de cinq ans d'expérience sur la base de données universelle DB2 ; il a écrit des articles dans de nombreux magazines et des livres sur DB2. Paul est largement certifié IBM : IBM Certified Advanced Technical Expert (DRDA et Cluster/EEE) et IBM Certified Solutions Expert

© copyright Éditions Eyrolles

sur la base de données universelle DB2. On peut le joindre à paulz_ibm@yahoo.com. Paul souhaite remercier Susan Williams et Darin McBride, développeurs IBM, de leur aide et leur passion pour DB2 et Linux.

Collaborateurs techniques

Chris Rogers a configuré des serveurs Linux depuis 1994. Pour donner une idée de ce que cela représente pour l'évolution de l'industrie informatique, le navigateur web était alors Mosaic, Netscape 0.9 avait tout juste atteint les sites FTP, et la distribution Slackware sans X-Windows tenait encore sur trois disquettes de 3.5 pouces seulement. Chris est maintenant CEO chez PowerStreak Networks, un fournisseur de services de gestion qui se spécialise dans la gestion des réseaux de clients et se sert de serveurs Linux. Chris peut être joint à crogers@powerstreak.com.

David C. Niemi est ingénieur réseau senior dans une grande société financière. Il est aussi président et co-fondateur de Tux.Org, une société sans but lucratif vouée à l'assistance et à la formation sur les logiciels libres. Depuis 1993, il apporte sa contribution au développement du noyau Linux et à d'autres projets de logiciels libres.

© copyright Éditions Eyrolles

Table des matières

INTRODUCTION 1

CHAPITRE 1
LE SYSTÈME D'EXPLOITATION LINUX 9

Le noyau Linux 11
L'avenir du développement de Linux 12
Distributions Linux 13
Pourquoi payer une version commerciale de Linux ? 13
Différences entre Linux et les autres systèmes d'exploitation 14
Fort potentiel 14
Multitâche 15
Architectures multiprocesseurs (SMP) 15
Mémoire virtuelle 32 bits (et 64 bits sur RISC) 15
Compatibilité binaire, le format ELF 15
Multi-utilisateur 16
Opérations en mode noyau complètement protégé 16
Interface graphique X Window gratuite 16
Applications Windows/DOS, VMWare 16
Fonctions réseau étendues 17
Services web standard 17
Services UNIX standard 17
Pourquoi choisir Linux 18
Quand utiliser Linux 18
Serveurs et stations de travail 18
Matériel recommandé 20

© copyright Éditions Eyrolles

Trouver les pilotes 20
Migrer vers des stations de travail Linux 20
Distributions de Linux 21
Revue rapide des distributions 21
Debian GNU/Linux 2.1 22
Caldera OpenLinux 2.3 22
TurboLinux 4.0 22
Linux Red Hat 6.1 22
Linux Mandrake PowerPack 6.1 23
SuSE Linux 6.3 23
Récapitulation des serveurs de base de données 23
SPIRO-Linux 24
Corel Linux 24
Réfléchir avant la mise à jour ou la migration 25
Compatibilité matérielle 25
Environnement de test 27
Mise à jour 27
Migrer depuis un autre système d'exploitation 27
Choisir une distribution 28
Comment entamer le processus 28
Formation de l'équipe 29
Solutions à présenter à votre directeur 29
Faire une analyse correcte des coûts 30
Résumé 30
FAQ 31

CHAPITRE 2
INSTALLATION DE BASE DE RED HAT LINUX 35
Machine physique séparée 36
Choisir une distribution Linux 37
Décisions préliminaires à l'installation 37
Matériel 37
Architecture multiprocesseurs 38
RAM 38
Disques 38

© copyright Éditions Eyrolles

RAID 39
Coût des systèmes RAID 39
RAID matériel ou logiciel 39
Interface(s) réseau 40
Mesures de sauvegarde 40
Assistance 41
Préliminaires à l'installation 41
Installer Red Hat 41
Un parcours d'installation rapide 41
Choisir un type d'installation 44
Installation d'une station de travail 44
Installation d'un serveur 45
Mise à niveau 46
Installation personnalisée 46
Installation personnalisée, ou manuelle 47
Créer les partitions de son choix 47
Partitions d'échange multiples 47
Supprimer une partition 47
Modifier une partition 48
Ajouter une partition 48
Installation du réseau 51
Utiliser DHCP 52
Activation au démarrage, ou boot 52
Ne pas utiliser DHCP 53
Hostname 53
Sélection du fuseau horaire 54
Configuration du compte root 54
Authentification 55
Utiliser NIS ou NIS+ 56
Sélection des paquetages 57
Les paquetages de base 57
Services optionnels à installer 58
Option « Tout installer » 60
Installation de X Window (optionnelle) 60
Logiciel serveur X 62
Modes vidéo X 62

© copyright Éditions Eyrolles

Installation réelle physique 62
Installation des paquetages 64
Installer Debian en jouant à Tetris 64
Linux Loader (LILO) et disquette de démarrage 64
Configuration de LILO 64
Disque de boot d'urgence 64
Installation terminée ! 65
Configurer le serveur pour une base de données 65
Points à vérifier 66
Autres versions de Linux 66
Exemples de scénarios de base de données 67
Résumé 68
FAQ 69

CHAPITRE 3
INSTALLER ET UTILISER ORACLE SUR LINUX 71

Concepts 73
Zone globale système 74
Pool partagé 74
Cache mémoire de la base de données 74
Cache de journalisation 75
Zone globale des programmes 75
Tablespaces 75
Fichiers de données 75
Extents 75
Segments 76
Fichiers de contrôle 76
Journaux 76
Segments de rollback 76
Fichiers de paramètres 77
SID 77
$ORACLE_HOME 77
$ORACLE_BASE 77
Identifiants de version 77
PL/SQL 78

© copyright Éditions Eyrolles

Schéma 78

Installation 78

Pré-installation 79

Configurer le système de fichiers 80

Configurer l'espace d'échange 81

Paramètres du noyau 82

Comptes et groupes système 82

Options d'environnement 83

Paquetages Linux 85

Java pour Linux 86

Se procurer le logiciel Oracle 87

Installer Oracle8 87

Installer les correctifs 90

Créer les objets de base de données 93

Installer Oracle8i 94

Installer les correctifs 98

Construire les scripts de création de base de données 99

Exécuter les scripts de création de base de données 104

Post-installation 105

Comptes SYS et SYSTEM 105

Configuration du fichier de paramètres 106

Démarrage et arrêt automatiques 106

Configuration du réseau 108

Utiliser Oracle8/8i 110

Démarrage et arrêt 110

Processus en arrière-plan 113

Créer un compte 114

*SQL*Plus* 116

Dictionnaire des données 119

Import/export 120

Logiciels tiers 122

Orasoft 122

SQLWork 123

ObjectManager 124

Orac 124

Perl/DBI 124

© copyright Éditions Eyrolles

Résumé 125

FAQ 126

CHAPITRE 4
INSTALLER INFORMIX SUR LINUX 129

Installation 135
Acquisition et mise en place du logiciel sur le serveur 135
Installation et marquage 138
Initialisation du disque et démarrage du serveur 141
Création des fichiers de données pour Informix 143
À propos des disques 144
Disques et répertoires 145

Besoins d'espace 145

Configuration 146
$INFORMIXDIR/etc/$ONCONFIG 147
$INFORMIXDIR/etc/sqlhosts 159
/etc/services 159
/opt/data/rootdbs 160
Démarrer votre moteur avec oninit 160
Arrêter le moteur 167

Configuration finale 167
Rappel 167
Physdbs 168
Créer un fichier de physdbs 169
Logsdbs 173
Créer logsdbs 174
Créer des nouveaux logs logiques 175
Tempdbs 184
Dernières options de configuration de $ONCONFIG 187
Log physique 187
Log logique 187
Utilisation de la mémoire 187
Options LRU 188
Options DBSPACETEMP 188
Redémarrer le moteur 189

© copyright Éditions Eyrolles

Autres outils 191
Dbaccess 194
Onmonitor 195
Ressources 196
Support technique Informix 196
Site web Informix 197
Groupe de news Usenet comp.databases.informix 197
Groupe international des utilisateurs Informix (IIUG) 197
Presse Informix 197
Formation Informix 198
Résumé 198
FAQ 199

CHAPITRE 5
INSTALLER ET UTILISER SYBASE SUR LINUX ... 201

Installation 203
Installer SQL Server 11.0.3 203
Configurer la machine de destination 204
Installer Sybase 206
Installer les logiciels clients optionnels 208
SQSH 209
Bibliothèques de développement 209
Configuration 210
Configurer le serveur de base de données Sybase 211
Configurer le serveur de sauvegarde Sybase 218
Configurer les bibliothèques client-serveur de Sybase 219
Démarrer automatiquement le serveur de base de données et le serveur de sauvegarde 220
Modifier le mot de passe de l'administrateur système 220
Configurer les unités de bases de données et les bases de données 221
Créer des comptes utilisateur et des autorisations 223
Tester la base de données 226
Conception d'une base de données 231

© copyright Éditions Eyrolles

Problèmes 233
Colonne Identité (auto-incrémentée) 233
Conformité à SQL 235
Environnement d'exécution 237
Utilisation de la mémoire par Sybase 237
Scripts rc de Sybase 238
Tempdb 239
Résumé 242
FAQ 243

CHAPITRE 6
INSTALLER DB2 VERSION 6.1 SUR LINUX RED HAT 245

Introduction 246
Préparer l'installation de DB2 sur une station de travail Red Hat 249
Préparer Red Hat version 5.2 et version 6.0 à une installation de DB2 249
Préparer Red Hat version 6.1 à une installation de DB2 250
Installer DB2 251
Exécuter l'installation 252
Vérifier l'installation 261
Configurer le centre de contrôle 263
Installer un client DB2 265
Configurer un client DB2 pour communiquer avec un serveur DB2 273
Dépannage de la connexion 280
Résumé 281
FAQ 282

CHAPITRE 7
INSTALLER MYSQL SUR LINUX 287

Installation 289
Conventions d'appellation 291

© copyright Éditions Eyrolles

Installation d'une distribution binaire 293
Installation d'une distribution RPM 294
Installation d'une distribution source 294

Configuration 296
Sécurité 297
Privilèges 298
Contrôle d'accès 302
Installation du système 303
Performance 305

Problèmes 308
Threads 309
Environnement d'exécution 309

Dépannage 311

Résumé 312

FAQ 314

CHAPITRE 8
INSTALLER ET GÉRER PROGRESS SUR LINUX 317

Installation 319
Installer à partir du média 319
Conditions préalables 319
Installer 320
Paramètres du noyau 324
Mémoire partagée 325
Sémaphores 326
Autres paramètres divers 327
Installation de l'environnement 328
Variables d'environnement 329
Installation du terminal — PROTERMCAP 329
Installation du mode client-serveur 330

Configuration 331
Structure de répertoire 331
Espace disque et débit d'entrée/sortie 333
Créer une nouvelle base de données 335
Multivolume 335

© copyright Éditions Eyrolles

Calibrage du cache 337

Exécuter Progress 339

Gérer la base de données 339

Démarrer la base de données 340

Démarrer une session 342

Arrêter la base de données 343

Sauvegarder la base 344

Surveillance 345

Dépannage 350

Utilisateurs suspendus, emballements et autres psychodrames 350

Forces et faiblesses 352

Le L4G 352

Orientation OLTP 353

Restauration après panne « à l'épreuve des balles » 353

Coût de propriété 354

Indexation des mots 354

Internationalisation 354

Fonctionnement 24 heures sur 24 355

Aucun BLOB 355

Pas de requêtes parallèles 355

Aucune gestion de verrouillage distribué 356

Résumé 359

FAQ 360

CHAPITRE 9

INSTALLER ET GÉRER POSTGRESQL SUR LINUX 363

PostgreSQL dans le sillage de l'Internet 365

Se procurer PostgreSQL 366

Installation rapide de PostgreSQL 367

Détails de l'installation 369

Installation de la distribution source 374

Avant de démarrer 375

Procéder étape par étape 375

© copyright Éditions Eyrolles

Valider l'installation de PostgreSQL 381
Les exemples de PostgreSQL 381
Tester l'interface Tcl/Tk 381
Tester l'interface Python — PyGreSQL 382
Tester l'interface Perl 382
Tester les interfaces libpq++ et libpq 383
Tester les interfaces Java 383
Tester l'interface ecpg 384
Tester l'interface ODBC 384
Tester l'interface MPSQL Motif-Worksheet 384
Tester les exemples SQL, les types définis par l'utilisateur et les fonctions 384
Vérifier l'installation de PostgreSQL 385
Appliquer les corrections d'urgence 386
Comment s'assurer de la fiabilité de PostgreSQL ? 386
Répertoires du système 387
Utiliser Postgres sur UNIX/Linux en environnement runtime 388
Démarrer postmaster 388
Les pg_options 389
Options définies 390
Sécurité 392
Authentification de l'utilisateur 392
Contrôle de l'hôte 393
Méthodes d'authentification 394
Authentification Kerberos 394
Disponibilité 395
Installation 395
Mise en œuvre 395
Créer des utilisateurs 396
Créer des groupes 396
Ajouter et supprimer des utilisateurs 397
Contrôle des accès 397
Fonctions et règles 398
Mises en garde 398
Sécuriser les connexions TCP/IP 398

© copyright Éditions Eyrolles

Exécuter une passerelle sécurisée via ssh 399
Gestion des disques et support des grandes bases de données 399
Localisation secondaire 400
Gérer une base de données 401
Créer une base de données 401
Accéder à une base de données 402
Détruire une base de données 403
Sauvegarde et restauration 403
Grandes bases de données 404
Utiliser un boîtier de commutation avec PostgreSQL 405
Échec au démarrage de postmaster 406
Problèmes de connexion du client 407
Messages de débogage 408
pg_options 408
Ressources PostgreSQL sur l'Internet 410
Support technique 410
Listes de diffusion 411
Comptes e-mail pour PostgreSQL 411
Listes de diffusion en anglais 411
Archives de listes de diffusion 412
Listes de diffusion en espagnol 412
Outils graphiques pour PostgreSQL 413
Pilotes ODBC, JDBC et UDBC 414
Interfaces Perl et DBI 415
Interface Perl 5 pour PostgreSQL 415
Perl DBI, de quoi s'agit-il ? 416
DBD-Pg-0.89, pilote DBI pour PostgreSQL 417
Support technique pour DBI 417
DBI, Perl et DBperl, de quoi s'agit-il ? 417
Spécifications DBI 418
Guides pour PostgreSQL 418
Ressources sur l'Internet 418
Sites de référence de PostgreSQL 420
Résumé 420
FAQ 421

© copyright Éditions Eyrolles

CHAPITRE 10
DÉVELOPPER UNE APPLICATION WEB 427

Plates-formes d'application web 428
Active Server Pages 429
Cold Fusion 429
Java Server Pages 429
Zope 429
Langages de script 430
PHP 430
Apache 430

Démarrage 431
Enjeux de la conception 431
Clarté 431
Extensibilité 432
Interface conviviale 433
Conception du schéma 433
Flux de données 433
PHP, MySQL et Apache : installer l'application 435
Connectivité de PHP : faire la connexion 436
Lire les données de la base 436
Sécurité 437

Conclusion 438

FAQ 439

ANNEXE A
SCRIPT PHP DE L'APPLICATION D'E-COMMERCE 441

ANNEXE B
SCRIPT MYSQL DE L'APPLICATION D'E-COMMERCE 465

INDEX 469

© copyright Éditions Eyrolles

Introduction

Comment croire que Linux a moins de dix ans ? Dans ce court laps de temps, le système d'exploitation, modeste, écrit à titre expérimental par un étudiant finnois, a explosé, qu'il s'agisse des fonctions ou de son usage. L'explosion de l'Internet, tout à fait comparable et simultanée, a propulsé Linux sur le devant de la scène.

Ces temps-ci, Linux bénéficie d'un gain de popularité, en tant que plate-forme serveur orientée entreprise particulièrement fiable. Depuis sa première publication, Linux s'est révélé un système d'exploitation stable et extensible. Mais ce n'est que depuis peu que les plus grands éditeurs de logiciels s'y intéressent et commencent à concevoir des applications portables sur les plates-formes Linux. L'avancée la plus remarquable dans ce domaine est venue en 1999, quand les quatre plus grands éditeurs de base de données — Oracle, Informix, Sybase et IBM — ont annoncé la prise en charge des plates-formes Linux. Auparavant, Linux disposait d'autres bases de données, mais la participation de ces éditeurs de renom lui donne le coup de pouce décisif et la publicité dont il avait besoin.

Les chapitres de cet ouvrage vous guident au fur et à mesure de l'installation de diverses bases de données, et indiquent aussi comment tester et utiliser certaines caractéristiques critiques pour les maintenir. Il appartient au lecteur de décider quelle base de données choisir pour un projet donné ; il trouvera dans ce livre quelques repères qui l'aideront dans sa démarche.

© copyright Éditions Eyrolles

Organisation de l'ouvrage

Ce livre traite de l'installation et de la configuration de plusieurs bases de données – parmi les plus grandes. Dans cette optique, chaque chapitre est écrit comme un tout et constitue une mini-référence pour la base de données traitée.

- **Chapitre 1 - Le système d'exploitation Linux.** C'est un résumé de Linux en général. Ce chapitre explore un peu l'histoire ainsi que certains aspects logistiques de Linux ; c'est un bon chapitre de fond pour se familiariser avec Linux et la mouvance des logiciels libres ou *Open Source*.
- **Chapitre 2 - Installation Red Hat Linux de base**. C'est une revue, étape par étape, de l'installation de Red Hat, la version la plus répandue de Linux. Si le serveur de base de données envisagé est destiné à être un serveur de production, il est vivement conseillé d'installer un système d'exploitation tout neuf, sur un nouveau matériel, afin d'obtenir la configuration la plus stable possible.
- **Chapitre 3 - Installer et exécuter Oracle sur Linux.** Oracle est en passe de devenir la base de données la plus courante et la plus largement reconnue du marché. Elle se prête à nombre d'extensions et est extrêmement fiable. Cependant, elle est assez chère et peut ne pas convenir aux applications des PME.
- **Chapitre 4 - Une installation Informix sur Linux.** Informix suit Oracle de près quant à la flexibilité et la fiabilité. En général, le choix entre les deux bases de données est strictement une question de préférence personnelle.
- **Chapitre 5 - Installer et utiliser Sybase sur Linux.** Sybase a un fidèle partisan. Quand on connaît Microsoft SQLServer, on sait d'entrée que Sybase constitue un choix logique pour l'environnement Linux : la conception de SQLServer, de Microsoft, est très proche de celle de Sybase.
- **Chapitre 6 - Installer la base de données universelles DB2 version 6.1 sur Linux.** DB2 est la plus sous-estimée des grandes bases de données. Elle est très souvent utilisée sur les grands systèmes, le portage sur Linux est donc puissant et flexible. Ne pas écarter DB2 des alternatives possibles sans lire attentivement ce chapitre.
- **Chapitre 7 - Une installation MySQL sur Linux.** MySQL est la plus connue des bases de données freeware. Elle est facile à initialiser et à configurer, et se prête très bien aux applications web. Cependant, elle est moins adaptée aux très grandes bases de données.

© copyright Éditions Eyrolles

- **Chapitre 8 - Installer et gérer Progress sur Linux.** Progress est habituellement négligé par la plupart des DBA. Il est souvent inclus dans des applications tierces, mais c'est en soi une très bonne base de données.
- **Chapitre 9 - PostgreSQL sur Linux.** PostgreSQL est disponible dans plusieurs distributions Linux. C'est un logiciel Open Source, comme MySQL, qui présente l'avantage d'être facile à démarrer dans le cadre d'une petite application de base de données. Cependant, sa flexibilité est limitée.
- **Chapitre 10 - Développer une application web**. On donne ici un exemple d'application des base de données au contexte du Web.

Les groupes Usenet en ligne, les livres blancs et les groupes d'utilisateurs sont des ressources intéressantes pour échanger des idées entre professionnels de l'informatique. Dans ce but, nous avons indiqué, tout au long du livre, des contacts et des ressources accessibles sur le web.

Bases de données sous Linux

Vous avez probablement déjà pris la décision d'implémenter une base de données sous Linux. La stabilité et la fiabilité du système d'exploitation Linux conviennent bien à une base de données. En tant que système d'exploitation Open Source, Linux dispose d'une bonne assistance et fait l'objet de mises à jour en permanence. L'utilisateur doit bien comprendre dès le début que les versions majeures du noyau et les corrections mineures de bogues doivent être appliquées régulièrement au système d'exploitation.

Il faut aussi être attentif au matériel qui héberge le serveur de base de données. Le traitement de la redondance y est intégré. Il est fortement conseillé de disposer de disques multiples dans une configuration de type RAID (*Redundant Array of Independent Disks*). Des bases de données différentes appellent des configurations adaptées, différentes. Il va de soi que le maximum possible de mémoire sera installé ; la meilleure base de données échouera misérablement si le matériel ne répond pas.

Le système d'exploitation et le serveur étant mis en place, plusieurs questions se posent :

- Que doit faire la base de données ?
- De quel environnement de production spécifique dispose-t'on : petit, moyen ou grand ?
- Est-ce une application critique ?

© copyright Éditions Eyrolles

- Cette application nécessite-t-elle de l'assistance ?

Les bases de données traitées dans ce livre ont chacune leur force et leur faiblesse. Si la décision n'est pas déjà prise, ou même si elle l'est, chacune sera prise en considération. Une réflexion supplémentaire ne nuit pas, tant que la solution n'est pas implémentée.

Oracle se situe toujours en tête, dans l'esprit de la plupart des gens, du point de vue de la souplesse et de la fiabilité. Mais Oracle requiert des administrateurs hautement qualifiés, ainsi que des contrats d'assistance avec la société distributrice. Cette combinaison, ainsi que le coût élevé des licences commerciales d'Oracle, exclut généralement les petites sociétés qui ont besoin de petites bases de données. Informix suit le même modèle. Il s'agit d'une base de données hautement fonctionnelle et fiable, mais ses coûts d'acquisition et d'assistance sont souvent, eux aussi, élevés.

Sybase, enfin, se présente comme un bon compromis, une solution viable. Même si elle ne peut prétendre proposer un ensemble de caractéristiques aussi riche que celui d'Oracle ou d'Informix, elle n'a certainement pas de lacunes sur l'essentiel. Son coût est plus accessible et, si l'on est familiarisé avec Microsoft SQLServer, sa configuration est immédiate. Sybase s'adapte à la plupart des applications et, en même temps, est stable. On peut dire la même chose de DB2, peut-être un peu plus délicate à maintenir, mais tout aussi riche en fonctions et susceptible d'évoluer qu'Oracle. Le coût des applications commerciales est, lui aussi, moins élevé. Progress entre dans cette catégorie.

Les bases de données en accès libre (*freeware*), telles MySQL et PostgreSQL, sont de très bonnes solutions d'entrée de gamme. Le coût en est très attrayant, bien sûr… elles sont gratuites. Ces bases de données sont faciles à installer et à maintenir. Un utilisateur novice, avec l'aide de ce livre, peut les installer et les démarrer en un jour ou deux seulement. Le principal inconvénient de ces bases de données est leur taille limitée : elles ont souvent des problèmes de performance au-delà d'un million d'enregistrements. Mais il existe une solution parfaitement viable : démarrer avec une base de données freeware, puis migrer vers l'une des plus grandes. Exporter les tables en SQL puis importer le SQL dans une autre base de données ne pose pas de problème. Passer du temps dès le début sur une solide conception de la base de données et sur un bon modèle de données évite de se mettre martel en tête. C'est vrai, bien sûr, pour toutes les implémentations de base de données, mais ce l'est particulièrement dans une situation de migration des données.

Appuyez-vous sur ce livre pour évaluer les solutions qui répondent le mieux à vos besoins, sachant qu'en dernier ressort, vous seul êtes à même de prendre une décision.

© copyright Éditions Eyrolles

Plus de bases de données pour plus de satisfaction

Les bases de données d'hier sont une pâle esquisse des applications puissantes produites aujourd'hui par les entreprises de pointe. Les décideurs et les administrateurs se trouvent maintenant face à une foule d'options, car la standardisation *via* SQL rend les données très portables. L'utilisation des BLOB (*Binary Large OBjects*, grands objets binaires) permet de ranger des images et des programmes dans des bases de données. La technologie XML (*eXtensible Markup Language*), relativement nouvelle, permet de présenter les données, qui étaient souvent verrouillées à un niveau de deux ou trois couches d'application, instantanément sur le web. D'une manière générale, les bases de données dont il est question dans ce livre peuvent se classer en deux catégories : traditionnelle et Open Source.

Bases de données traditionnelles

Les bases de données traditionnelles sont, en principe, programmées par de grands éditeurs de logiciel et font l'objet d'une assistance. L'avantage le plus sensible de ce modèle est le support de produit. Pour un certain coût de maintenance annuelle, l'éditeur met à disposition le dépannage et la mise à jour du produit. Les très grandes entreprises dont les systèmes d'information reposent sur de grandes bases de données se refusent absolument à maintenir les couches sous-jacentes, ou annexes, ce que ce modèle de support propose.

L'inconvénient évident du modèle traditionnel est la lenteur de l'éditeur à modifier ses produits et à lui ajouter de nouvelles caractéristiques. Il faut souvent des mois, voire des années, de développement pour que les technologies les plus récentes parviennent à l'utilisateur final. Autre inconvénient majeur : ces sociétés, comme n'importe quelle entreprise, sont obligées d'adopter une tactique commerciale entraînant la hausse du coût des licences individuelles, qui deviennent prohibitives pour beaucoup d'utilisateurs. L'éditeur est habituellement l'objet d'une concurrence intense, de sorte que les produits apparaissent trop tôt et que des corrections sont souvent nécessaires. Prenons l'exemple d'Oracle 8i : cette version longuement attendue d'Oracle a été programmée pour être compatible avec l'Internet, mais elle est pleine de bogues. Ces bogues ont été corrigés dans la deuxième version d'Oracle 8i, qui ne fut publiée qu'un an plus tard. C'est le scénario avec lequel beaucoup d'éditeurs jonglent en permanence.

© copyright Éditions Eyrolles

Bases de données Open Source

Le modèle de développement de logiciel le plus récent est l'Open Source. Il a le grand avantage de reposer sur une communauté de développement. Il y a normalement un ou des responsables du code en nombre restreint, mais tout l'Internet comme équipe de développement potentielle. Le logiciel qui en résulte est revu et mis à jour souvent. La version la plus récente d'une certaine partie du logiciel hérite des caractéristiques et de la technologie les plus récentes. Cela, bien sûr, a pour conséquence fâcheuse que ces caractéristiques peuvent ne pas avoir été entièrement testées. L'administrateur système doit être vigilant quant aux nouvelles mises à jour et additions, et il faut un utilisateur avancé pour maintenir le système. Les grandes entreprises se heurtent au fait que l'on ne dispose pas de l'assistance et de la compatibilité traditionnelles — l'assistance incombe à l'administrateur. En toute honnêteté, c'est exactement de cette manière que cela fonctionne dans la plupart des sociétés. L'administrateur fait appel à ses propres connaissances et s'appuie sur les informations obtenues dans les sites web et les groupes de news relatifs au logiciel en question.

Utiliser au mieux sa base de données

Avoir une base de données sur un serveur Linux est une bonne chose, mais que peut-on en faire ? Pour tout un chacun, la réponse est : « Dieu reconnaîtra les siens ». Gérer les données du personnel, les contraintes clients, corréler des données marketing, améliorer la performance d'un système de commandes, et bon nombre d'autres applications seront conçues et programmées.

Caractéristiques de base

Toutes les bases de données parlent le langage SQL (*Structured Query Language*). Ce standard de base assure la compatibilité d'une base de données à une autre. Mais il n'est pas implémenté de la même manière d'une base de données à une autre, aussi faut-il être attentif lors d'une migration. Par exemple, n'essayez pas de migrer une base de données Oracle utilisant les BLOB dans une base de données PostgreSQL, qui ne gère pas ce type d'objets. Toutes les bases de données supportent les quatre instructions SQL de base : SELECT, INSERT, UPDATE et DELETE et, de plus, elles supporteront toutes les notions de base utilisateur et de schéma de table, qui servent à créer et gérer les tables et les autorisations des utilisateurs.

© copyright Éditions Eyrolles

Caractéristiques avancées

Les deux caractéristiques avancées les plus notables sont les BLOB et XML. Les BLOB (grands objets binaires) permettent de stocker des images dans la base de données. Oracle et d'autres bases de données supportent les BLOB. XML est une technologie relativement nouvelle. Semblable sur ce point à HTML, XML est un sous-ensemble de SGML (*Standard Generalized Markup Language*). Mais, contrairement à HTML, qui est un langage à balises dédié au formatage de documents, XML contient des balises relatives aux données et permet d'afficher les données directement dans une page web. Si le navigateur ou le serveur d'application « parlent » XML, les données sont formatées et affichées avec un minimum de programmation. La plupart des bases de données disponibles sur le marché aujourd'hui prennent maintenant en charge XML d'une manière ou d'une autre.

Les bases de données, leur avenir

Le web a explosé au cours des trois dernières années. Utiliser des bases de données sur le web est d'ores et déjà entré dans les moeurs. Stocker des informations sur les clients et les utilisateurs, accéder aux profils personnels, trier et cataloguer des données et des informations sur les produits accessibles aux clients est en passe de devenir le nouveau standard — le modèle qui pilotera le e-commerce. La recherche approfondie de données, les portails et les commandes par correspondance sont au coeur de négociations en cours, concernant l'avenir des CEO.

Dans le cadre de ce livre, nous avons présenté un scénario d'achat à la carte (commandes par correspondance), pour illustrer l'une des utilisations possibles des bases de données. Les serveurs web sont un élément clé de l'utilisation des bases de données sur le web, et nous en présentons un bref aperçu. Nous espérons surtout que vous tiendrez ce livre pour une référence utile, susceptible de vous aider à mettre en place une base de données sur Linux, quel qu'en soit l'usage.

A qui s'adresse cet ouvrage ?

Ce livre est, en tout premier lieu, destiné aux administrateurs de base de données et aux utilisateurs de Linux soucieux d'améliorer leurs compétences et leur compréhension des diverses bases de données existantes. Nous espérons que vous fermerez ce livre sur une meilleure compréhension de la

© copyright Éditions Eyrolles

conception des bases de données et de leur implémentation en général. Un strict ensemble d'instructions, à exécuter de manière quasi mécanique, pour installer et configurer une base de données a, certes, son intérêt, mais ce livre tente d'aller au-delà : vous faire mieux comprendre le système d'exploitation Linux, quand il est utilisé comme serveur de base de données.

© copyright Éditions Eyrolles

1

Le système d'exploitation Linux

Dans ce chapitre

- Brève histoire de Linux
- Le noyau Linux
- La différence entre Linux et les autres systèmes d'exploitation
- Pourquoi choisir Linux ?
- Distributions de Linux
- Réfléchir avant la mise à jour ou la migration
- Comment commencer le processus

© copyright Éditions Eyrolles

Brève histoire de Linux

Le système d'exploitation Linux n'a cessé de se développer depuis sa création, en 1991 : Linus Torvalds commença alors à écrire un système d'exploitation, fondé sur UNIX, destiné à fonctionner dans l'environnement PC Intel.

Diplômé de l'université de Helsinki, en Finlande, Linus vit l'intérêt que présentait un PC de base, en tant que système de bureau, par rapport aux terminaux passifs. En ce temps-là, le coût du logiciel de bureau et du système d'exploitation associé interdisaient à l'étudiant ou à l'utilisateur moyen de disposer d'un PC équipé des produits (logiciels) nécessaires. Dans le but de résoudre ce problème, et d'augmenter les ressources de l'école où il travaillait, Linus se mit au travail et décida de redévelopper le code Minix (un clone d'Unix microkernel) pour créer quelques utilitaires simples, pour lui-même. Il sentit que cela en valait la peine, même si, dans un premier temps, cela ne servait qu'à permettre à l'école d'utiliser quelques anciens PC comme terminaux passifs pour se connecter à des serveurs (principaux) maîtres.

Linus, à la recherche d'idées de développement et de ressources, s'ouvrit de son projet à la communauté Internet. Début 1992, seulement une centaine de personnes avaient utilisé Linux. Le noyau Linux originel était très compact et ne proposait que les fonctionnalités de base. La communauté Internet s'est vite mobilisée et a soutenu sans défaillance le démarrage du projet. Très vite, des centaines, puis des milliers de programmeurs ont spontanément proposé leur collaboration, ainsi que des enrichissements sous forme de développements personnels. Cela a permis un développement très rapide, qui a mis à profit beaucoup de bonnes volontés pour déboguer et tester le code de Linux. En peu de temps, Linus a remplacé tout le code Minix par son propre code, enrichi des contributions des autres volontaires. En s'appuyant sur la communauté *ppen-source*, qui était encore dans sa prime jeunesse, Linus a livré Linux au monde entier.

En termes de développement d'utilitaires, il n'y avait pas, et il n'y a toujours pas, de « meilleure conception » d'un utilitaire. N'importe qui peut développer ce qu'il estime être un utilitaire réellement utile. Il est évident que, dans les faits, on trouve pour la même fonction de base, plusieurs utilitaires ; beaucoup sont conçus de sorte que, si vous êtes déjà familiarisé avec un utilitaire, Linux vous en fournira un complètement équivalent du point de vue fonctionnel. Par exemple, il existe au moins une douzaine d'éditeurs de fichiers disponibles, tant en version X Window qu'en version texte.

Près d'une décennie plus tard, Linux commence à trouver sa place dans la communauté professionnelle en tant que moyen viable, et souvent supérieur, de remplacer beaucoup d'autres systèmes d'exploitation de bureau et de serveurs. Linux a pris d'assaut dans un domaine nouveau — celui des

© copyright Éditions Eyrolles

serveurs de bases de données — ainsi que le prouvent les nombreuses alliances entre les divers éditeurs de Linux et certains des grands éditeurs de bases de données, tels Oracle ou IBM. Cet ouvrage se propose de présenter les forces de Linux en tant que plate-forme serveur de base de données, d'examiner plusieurs solutions de base de données, et de décrire quelques exemples d'applications réelles.

Le noyau Linux

Au départ, Linux fut développé pour un usage récréatif et il était destiné à être distribué gratuitement. Le noyau Linux et son code source sont disponibles pour tous et continueront à être gratuits tant que Linus maintiendra le copyright sur le code source du noyau. Linux est une marque commerciale déposée par Linus Torvalds et est écrit collectivement, aux termes de la licence générale publique (GPL, *General Public License*), par les centaines d'auteurs qui contribuent à sa publication. La GPL autorise n'importe qui à distribuer librement le code source dans son intégralité, à en faire des copies, et à le vendre ou à le distribuer (à quelques exceptions près) par tous moyens. La GPL est incluse dans le noyau Linux (répertoire **usr/src/linux/COPYING/** sur le CD-ROM Linux), ou peut être téléchargée, ou consultée, sur de nombreux sites Internet.

Important. Un logiciel sous GPL a le mérite d'offrir une totale souplesse aux utilisateurs du logiciel, et de leur épargner l'achat d'une licence chère ou des poursuites.

Le noyau Linux n'est qu'une petite partie de ce que l'on connaît sous le nom de Linux. Le noyau, comme son nom l'indique, est la pièce centrale, au coeur du système d'exploitation. Vous pouvez penser au noyau comme au moteur de votre PC : ce qui « fait tout marcher ». Le noyau gère les entrées/sorties, se charge de la gestion de la mémoire... c'est, à la base, le cerveau du système d'exploitation. Le noyau, bien que compact, est l'une des raisons majeures qui font que Linux est un système d'exploitation puissant et efficace.

En plus de cette partie centrale, il faut encore beaucoup d'autres composants pour bâtir un système d'exploitation complet et utilisable. La puissance d'un système d'exploitation ne peut être complètement testée sans utilitaires pour accomplir des travaux tels que la création de documents, le partage de fichiers et l'installation, l'affichage et l'impression de fichiers et de documents. Ces utilitaires sont disponibles librement sur l'Internet, de même que le noyau de Linux. Le tableau 1.1 présente un exemple restreint des utilitaires fournis avec Linux.

© copyright Éditions Eyrolles

Tableau 1–1
Utilitaires UNIX de base fournis avec chaque distribution Linux

Opération	Ligne de commande ou fichier
Manipulation de fichier	cp, mv, rm, ln, touch, ls, cat, more, >, >>, 2>, <
Installation	ifconfig, mount, showmount, /etc/printcap, /etc/hosts
Statistiques	ps, vmstat, nfsstat, innstat, netstat, top, qui, users, pac, du
Utilitaires	vi, pine, pico, sed, awk, grep, find, netscape, tar, cpio, compress
Services réseau	DHCP, http, SMB, ftp, telnet, talk, pop3, imap, fingerd, sendmail

Ces utilitaires, ainsi que des centaines d'autres, sont disponibles librement sur l'Internet. On peut les télécharger et les installer individuellement, ou télécharger le noyau et tous les utilitaires nécessaires dans un fichier unique (quoique de bonne taille). À la place du téléchargement, beaucoup d'éditeurs de logiciels offrent leurs versions de Linux sur CD-ROM, que l'on peut acheter dans n'importe quel magasin. Ces versions commerciales comprennent un ensemble de programmes et d'utilitaires, ainsi que quelques utilitaires maison : c'est ce que l'on appelle une version ou *distribution* de Linux.

L'avenir du développement de Linux

Le code source est public : même si Linus décide d'arrêter de travailler, quelqu'un d'autre peut facilement prendre le code source et poursuivre le développement. Et le noyau n'est qu'une petite, quoique très importante, partie de Linux. La plupart des centaines d'utilitaires ont été réalisés dans le cadre du projet GNU ou émanent de l'*open-source*.

Linus n'est pas l'unique développeur du noyau au monde. Beaucoup de gens ont été mis à contribution à ce jour, et d'autres le seront encore. Alan Cox, notamment, a contribué au développement du noyau dans des domaines tels que la gestion des nouveaux matériels ou la prise en compte des contraintes des nouveaux processeurs. À titre d'exemple de communauté d'intérêt, AMD fournit tous les détails du processeur Athlon à Alan Cox. Les noyaux réalisés dans un avenir très proche doivent atteindre des vitesses record, si l'on veut les optimiser pour fonctionner sur une machine Athlon.

© copyright Éditions Eyrolles

Distributions Linux

Une copie de Linux, commerciale ou gratuite, s'appelle communément une *distribution*. Il s'agit d'un ensemble d'utilitaires pour Linux, fourni par un éditeur, comprenant le noyau et parfois certains produits propriétaires, tels que des gestionnaires de base de données, des logiciels pare-feu, etc. Une distribution typique de Linux contient des centaines d'utilitaires UNIX standard, et aussi un excellent ensemble d'outils de développement, de jeux, de bases de données, de logiciels graphiques et d'utilitaires de contrôle, ainsi qu'une interface graphique (GUI, *Graphical User Interface*), le système X Window.

Certaines de ces distributions Linux ne sont pas entièrement gratuites. La base du système peut être gratuite, mais non les services supplémentaires qui font l'originalité d'une version de Linux. Il existe des versions commerciales de Linux, produites par des éditeurs tels que Debian, Red Hat, VA Linux, SuSE, et d'autres. Selon les termes de la licence générale publique, on peut télécharger gratuitement une copie complète de ces distributions Linux. On les trouve habituellement sur le site web de chaque société, ainsi que sur de nombreux sites miroir dispersés à travers l'Internet.

Pourquoi payer une version commerciale de Linux ?

Les versions commerciales offrent des bonus supplémentaires, non disponibles dans les versions gratuites, bien que le support physique ait l'air identique. La différence se situe d'habitude dans quelques produits ou services à valeur ajoutée, compris seulement dans le produit commercial. Par exemple, Red Hat Linux, quand on l'achète auprès d'un distributeur, comprend de 30 à 180 jours d'assistance gratuite à l'installation, ainsi qu'une souscription au service central web de mise à jour. La distribution Debian propose une assistance illimitée pour ses sites ftp et web. On peut aussi acheter ces services d'assistance et de mise à jour sur la base d'un contrat annuel chez Red Hat ou d'autres fournisseurs. C'est de cette manière que Red Hat gagne de l'argent et investit dans le développement des fonctionnalités nouvelles de ses produits.

Les versions commerciales sont censées être maintenues, et proposer habituellement une procédure simplifiée d'installation et une interface unifiée. À ses débuts, Linux donnait un ensemble d'instructions de base pour installer la distribution Linux. Il fallait alors configurer fichiers, imprimantes, cartes réseau et cartes son individuellement. Les versions actuelles ont toujours au moins une interface d'installation, parfois en couleur, et à base de menus (les plus courantes), ou sont en mode graphique (X Window), comme la version courante de Red Hat.

© copyright Éditions Eyrolles

Différences entre Linux et les autres systèmes d'exploitation

Linux est un clone d'UNIX, écrit pour être compatible avec POSIX, partout où c'est possible. Linux a d'abord été écrit pour la plate-forme Intel, puis il a été porté sur PowerPC, Macintosh, Amiga, quelques ordinateurs Atari, et même sur le Palm Pilot. Si vous connaissez les commandes de base, l'organisation du système de fichiers et les utilitaires généraux de toute autre variante d'UNIX, vous les connaissez sur Linux. Si vous êtes utilisateur de Microsoft Windows ou de Macintosh, vous ne serez peut-être pas à l'aise au début, mais vous verrez vite que tout y est similaire.

Fort potentiel

Les distributions de Linux n'ont pas exactement le même ensemble d'utilitaires, mais en général, les fonctionnalités de base fournies avec Linux sont pour le moins impressionnantes. Comme Microsoft n'a pas besoin de se mesurer à la concurrence en matière de bureau, ses conseils font presque toujours référence à une amélioration par rapport à une version précédente, et non à un produit de la concurrence. Linux est un système d'exploitation compétitif pour Intel ou les compatibles PC tant comme station de travail que comme système d'exploitation d'un serveur. En fait, la plupart des améliorations de Windows NT, depuis quatre ans, ont été des services UNIX que Linux a toujours supportés.

Voici quelques-unes des caractéristiques de la plupart des variantes de Linux, qui en font un grand serveur de base de données :

- multitâche
- SMP (multiprocesseurs)
- modèles de mémoire virtuelle 32 et 64 bits
- mode noyau protégé
- binaires standard ELF
- multi-utilisateur
- interface graphique gratuite
- compatibilité binaire avec les autres variantes d'UNIX
- protocoles de réseau étendu et support des services
- tous les services réseau UNIX standard, tels HTTP, FTP, Telnet, etc.

© copyright Éditions Eyrolles

Multitâche

Le multitâche est une fonction essentielle, prérequise, de tout serveur, en particulier les serveurs de base de données. Avec un véritable environnement multitâche Intel 32 bits (64 bits sur les processeurs de type RISC), Linux a la capacité d'exécuter des programmes multiples « simultanément », même sur un processeur unique. En réalité, ce processus est un tour de table très rapide — c'est-à-dire que chaque processus attend son tour dans une file pour disposer des ressources. La méthode particulière dont se sert Linux pour gérer les requêtes donne l'impression d'un vrai multitâche. Plusieurs threads peuvent être démarrés simultanément et s'exécuter en parallèle, ce qui offre des performances excellentes et optimise l'usage des canaux d'entrée/sortie.

Architectures multiprocesseurs (SMP)

Sorti de la boîte, Linux prendra en charge jusqu'à seize processeurs, selon les variantes. Cette exploitation du multiprocesseur fait de Linux une plate-forme excellente pour un serveur de base de données, capable de gérer simultanément beaucoup de threads. La plupart des grands éditeurs vendent des serveurs multiprocesseurs spécifiques des bases de données et des services web. Le noyau Linux est bien conçu pour gérer ces processeurs multiples : il propose une utilisation quasi-complète de tous les processeurs supplémentaires, au lieu d'un ratio d'utilisation qui décline à chaque ajout d'un nouveau processeur.

Mémoire virtuelle 32 bits (et 64 bits sur RISC)

Linux s'appuie sur tous les services de gestion de la mémoire en 32 bits du processeur pour fournir une mémoire virtuelle de 4 Go sur les plates-formes Intel, et de 16 exaoctets[1] quand on se sert de Linux 64 bits sur un processeur RISC. Chaque nouveau processus crée un processus utilisateur indépendant avec 2 Go d'espace virtuel (disponible s'il le faut ; en réalité, il n'utilise que la RAM nécessaire pour le support du processus). Cela procure sécurité et indépendance entre tous les processus du système.

Compatibilité binaire, le format ELF

Linux a adopté une stratégie de conception d'applications fondée sur les formats binaires standardisés ELF (*Executable and Linking Format*) et les bibliothèques partagées, similaires aux bibliothèques de liens dynamiques (DLL), mais non compatibles. Cela permet d'exécuter les applications sur des systèmes d'exploitation multiples sans les recompiler. SCO UNIX 4.2, SVR4,

1. 1 exaoctet = 1024 petaoctets = 1024 * 1024 teraoctets = 1024*1024*1024 gigaoctets

© copyright Éditions Eyrolles

AIX, Sun, et la plupart des versions de Linux pour Intel figurent parmi les nombreuses variantes d'UNIX qui migrent vers cette conception de la compatibilité. Par exemple, Corel Draw pour SCO fonctionne aussi sous Linux.

Multi-utilisateur

De vraies fonctions multi-utilisateur permettent à plusieurs utilisateurs d'exécuter des processus multiples sur le même serveur hôte Linux. À ne pas confondre avec la « possibilité d'accès multiple », que propose Microsoft Windows NT, où des processus et des threads multiples s'exécutent, mais sans contexte d'utilisateur distinct et à part entière.

Opérations en mode noyau complètement protégé

Le système d'exploitation Linux exécute toutes les applications en mode noyau complètement protégé (c'est-à-dire qu'aucun processus ne peut circonvenir le système d'exploitation). Aucune application utilisateur ne peut interférer avec une autre, sauf si l'autorisation a été spécifiée, et il n'y a aucun accès direct aux matériels dans Linux. Ce sont les deux principaux problèmes rencontrés avec les systèmes d'exploitation de type DOS, qui les rendent particulièrement vulnérables aux virus et à d'autres problèmes de sécurité.

Interface graphique X Window gratuite

L'interface graphique utilisateur de Linux est totalement gratuite. Elle repose sur une version du standard XFree86 du système X Window (la plupart des programmes compatibles X existants fonctionneront sous Linux sans aucune modification). Le système X Window existe depuis le début des années 1980, précédant la conception de Microsoft Windows. Vous pouvez même remarquer que le nouveau « look » de Win9x/NT ressemble beaucoup au standard X Window, dans lequel nous avons baigné bien plus longtemps.

Applications Windows/DOS, VMWare

Avec la plupart des versions de Linux, sont fournis un paquetage d'émulation de DOS appelé DOSEMU et un paquetage d'émulation de Windows, WINE : ils fournissent un support de base aux applications « propres », qui n'essaient pas de forcer les règles en accédant directement au matériel ou en utilisant des API non documentées.

Un autre bon point pour Linux : le paquetage d'émulation de machine virtuelle appelé VMWare offre un support complet de Windows NT, 9.x et 3.x, qui permet à Linux d'être un système d'exploitation hôte pour un ou plusieurs de ces systèmes d'exploitation hôtes installés dans un monde virtuel, partageant avec Linux la mémoire et les ressources matérielles locales

© copyright Éditions Eyrolles

également et simultanément. On a littéralement à la fois Linux et un ou plusieurs systèmes d'exploitation Windows fonctionnant en même temps sur la même machine, dans des fenêtres X Window différentes ! On peut donc fournir l'un ou l'autre de ces bureaux à tout utilisateur du réseau équipé d'un PC relativement puissant (c'est-à-dire avec le matériel approprié : 266 MHz ou plus, 96 Mo RAM ou plus, et l'espace disque en quantité suffisante). À noter que VMWare offre un produit de machine virtuelle équivalent pour NT. On installe Linux comme système d'exploitation hôte sur NT et on exécute les deux bureaux ensemble à partir d'une fenêtre Windows. On peut télécharger, depuis l'adresse www.vmware.com, une version d'essai gratuite de 30 jours, pour l'un ou l'autre des systèmes.

Fonctions réseau étendues

Les fonctions réseau sont sans doute le meilleur argument de vente de la plupart des versions de Linux. Il comprend des protocoles natifs et une bonne assistance au niveau application pour presque tous les autres systèmes d'exploitation réseau couramment utilisés sur les matériels PC. En outre, il y a un support média étendu pour divers types d'interfaces, autres que l'Ethernet standard et Token Ring, tels que ISDN et même Ham Radio.

Un support complet des protocoles natifs est fourni pour Macintosh, DOS, Windows NT/9x/3.x, Novell et OS/2 en réseau, sous forme de paquetages complémentaires divers, ce qui fait que Linux se présente comme un serveur ou un client local natif à ces autres systèmes d'exploitation.

Services web standard

Le support des services Internet est fourni par le serveur web Apache, le produit de serveur web dominant, qui exécute les services http sur l'Internet, ainsi que des versions complètes des dernières interfaces CGI : Perl, Java, etc. Linux est un excellent serveur web ; Apache est une application web de haut niveau qui a intégré des hooks pour toutes les grandes bases de données utilisant ces interfaces CGI. La seule interface web à ne pas être proposée est Active Server Pages (ASP). Le support d'ASP est disponible chez de nombreux éditeurs tiers.

Services UNIX standard

Linux supporte tous les services UNIX standard : e-mail, FTP, news UseNet, Telnet, POP, IRC, NFS, DNS, NIS, SNMP, Kerberos, etc., ce que l'on attend de tout système Internet. Pour une vue d'ensemble beaucoup plus précise de ce qu'est Linux, consulter le site web international de Linux à http://www.li.org/li/index.shtml.

© copyright Éditions Eyrolles

Pourquoi choisir Linux

Les systèmes d'exploitation sont tous soumis aux mêmes contraintes de base : fiabilité, sécurité, fonctionnalités et coût. Si l'on prend tout cela en compte, Linux est difficile à battre. Peu d'autres systèmes d'exploitation peuvent se vanter de l'ampleur des applications, des pistes d'audit de fiabilité et des caractéristiques réseau les plus récentes pour un coût faible ou nul. Le logiciel initial est gratuit. Les coûts d'assistance courante et d'exploitation ne sont pas différents des coûts normaux ; par contre, il faut l'installer, le gérer et le maintenir. Le coût total de propriété est très bas. Linux, une fois installé et opérationnel, demande peu de surveillance et de maintenance : heureusement, car les administrateurs système qualifiés sont difficiles à trouver !

Quand utiliser Linux

Les usages les plus courants de Linux sont nombreux : équiper un routeur, le protocole d'adressage IP dynamique DHCP, le Web, le protocole de transfert de fichier (ftp), une base de données, un serveur de fichiers ou d'impressions. Il peut donc faire bonne figure dans les bureaux.

Linux est un serveur de pare-feu réputé, aux dernières normes de sécurité en matière de filtrage avancé de paquets, ainsi que d'encapsulation TCP, apte à fournir la traduction d'adresses réseau (NAT) ainsi que les services de proxy pour les hôtes intranet qui doivent accéder à l'Internet sans risque. Avec Apache et les services WS-FTP, vous disposez d'un site ftp et web très sûr, robuste, doté de la sécurité, des vérifications et des caractéristiques de gestion adéquates, qui dépassent tous les autres produits concurrents sur le marché. Diverses versions de Linux ont donné lieu à des alliances avec les plus grands serveurs de base de données, dans le but d'assurer une intégration et des performances optimales. Une telle alliance s'est faite entre Red Hat et Oracle, et il y en a d'autres (voir les détails sur le site web de chaque version).

Serveurs et stations de travail

Il n'y a pas de différence réelle entre une version station de travail et une version serveur. Dans l'un ou l'autre de ces emplois, Linux est toujours le même système d'exploitation. Cependant, Linux peut être conditionné et distribué sous différentes formes qui répondent à cette classification ; par exemple, la version station de travail TurboLinux se vend au détail moins cher que le paquetage serveur TurboLinux, bien que la seule différence réside dans le nombre et le type de logiciels que l'on peut installer. Une version station de travail comprend en général les principaux services réseau de Linux

© copyright Éditions Eyrolles

qui appartiennent à un utilisateur et des applications complémentaires orientées production, par exemple un ensemble de produits bureautiques. Une version serveur peut ne pas comprendre, par défaut, d'applications de ce type, mais ne doit pas vous empêcher de les installer plus tard.

Si l'on se sert principalement de l'hôte Linux pour des services de fichiers, d'impression, Web et ftp, cet hôte sera considéré comme un serveur. Si l'on y ajoute une suite bureautique, telles Corel WordPerfect, StarOffice ou Applixware (toutes équivalentes à Office 97), on obtient alors une station de travail. Cette désignation « station de travail » ou « serveur » est plus conforme à une perception de l'usage qu'à une restriction sur les fonctions du logiciel. La plupart des versions de Linux permettent aussi bien des applications serveur qu'utilisateur, et il faut sélectionner ce que l'on veut installer à chaque fois. On peut même automatiser l'installation avec une simple disquette de démarrage qui se connecte à un service central d'installation et contient un script identifiant les partitions de disque à créer, leur taille et les paquetages à installer. La distribution Red Hat propose le service d'installation automatisée Kickstart pour effectuer des installations multiples de Linux Red Hat sur des machines similaires.

Incompatibilités étranges

Récemment, il nous est arrivé de travailler avec quelques machines locales, qui contenaient une carte d'interface réseau courante, supposée fonctionner avec Linux. Hélas, ces cartes sont à l'origine d'une véritable frustration ; après vérification rapide de la page web du fabricant de la carte, qui nous avait donné une lueur d'espoir, les FAQ, dans une section spécifique de Linux, contenaient une suggestion qui a nécessité de télécharger des fichiers utilitaires d'installation de DOS. Le site indiquait que « Avec Linux, ces cartes fonctionnent mieux si l'on n'est pas en mode Plug and Play ». L'utilitaire d'installation, en tous cas, n'a rien changé du tout à cet état de fait, bien que ces cartes particulières fonctionnent bien sous Windows et NT. Aucune valeur de coaxing ou begging n'a convenu. Nous avons fini par les remplacer par des cartes 8 bits vraiment anciennes qui ont fonctionné au premier essai. On n'est donc jamais sûr de rien.

Le site web du fabricant indiquait que ses cartes réseau les plus récentes étaient vraiment compatibles Linux, et qu'il existait même quelques pilotes Linux pour elles. Heureusement, Linux est en train de devenir un nouvel OS avec lequel il faut compter, et les fabricants de matériel vont passer de plus en plus de temps à assurer sa compatibilité.

© copyright Éditions Eyrolles

Matériel recommandé

Les serveurs Linux fonctionnent bien sur les équipements anciens et sont un bon moyen d'en prolonger l'usage en tant que pare-feu, serveurs de fichiers et d'impression (nfs, SMB, Novell, Mac), serveurs ftp ou web. Mais pour l'activité de bureau, les modèles de ces dernières années sont les meilleurs. On n'a pas toujours besoin du matériel le plus récent et le plus puissant : souvent, il est très spécifique de Windows, comme WinModems. En général, les périphériques de marque ou les périphériques qui utilisent des puces connues de Linux fonctionnent le mieux. Linux a suffisamment de succès pour que l'on puisse se procurer facilement un pilote d'équipement courant sur l'Internet.

Trouver les pilotes

Le meilleur endroit où commencer à chercher les pilotes est le site web de la distribution Linux ; par exemple, www.RedHat.com, une petite partie du site web principal de Red Hat, ou www.debian.org/distrib/ftplist, une petite partie du site web principal de Debian. Vous pouvez aussi utiliser votre moteur de recherche favori dans votre navigateur web et taper les mots-clés « Linux » et « le nom de votre matériel ». Vous obtenez de nombreux liens vers les sites contenant des pilotes et avez des chances de trouver quelques favoris tels que des cartes vidéo, des cartes réseau et des cartes SCSI. La plupart des sites offriront aussi leur HOWTO la plus récente sur la compatibilité matérielle. En général, l'Internet sera votre meilleure source de documentation, de mise à jour et d'assistance pour Linux.

Migrer vers des stations de travail Linux

Chaque fois qu'une entreprise change de système d'exploitation, la migration prend un caractère traumatisant. On doit même recycler certains utilisateurs dans certains cas. Les compétences requises sur l'interface graphique resteront les mêmes, mais les applications peuvent se présenter différemment sur certains points, par exemple la suite bureautique. Mais si le bureau est un client dédié à une base de données, l'aspect et la perception seront similaires, et la fiabilité très élevée. Pour vos utilisateurs finals, vous pourrez examiner la suite bureautique StarOffice, fournie gratuitement par Sun Microsystems — voir www.sun.com/products/staroffice/get.cgi. StarOffice est couplée avec de nombreuses versions de Linux, dont Red Hat 6.1 et TurboLinux 4.0. Vous pouvez les trouver, ainsi que d'autres produits, tels les nouveaux produits de la suite GNU, en visitant des sites comme www.slashdot.org et www.freshmeat.net : vous y trouverez des nouvelles de la communauté Linux et pourrez souvent télécharger les plus récents logiciels de jeux, des applications, des correctifs et des mises à jour, pour toutes les versions de Linux.

© copyright Éditions Eyrolles

Distributions de Linux

Il existe de nombreuses distributions de Linux, qu'elles soient commerciales ou non. Le tableau 1.2 récapitule les plus connues.

Tableau 1–2
Liste non exhaustive des distributions Linux disponibles

Site web	Distribution
www.caldera.com	Caldera Systems OpenLinux
www.conectiva.org	Conectiva Linux
www.debian.org	Debian GNU/Linux
www.libranet.com	Libranet
www.linux-mandrake.com	Linux Mandrake
www.linuxppc.com	LinuxPPC (PowerPC)
www.phatlinux.com	Phat Linux
www.RedHat.com	Red Hat Linux
www.slackware.com	Slackware Linux
www.stampede.org	Stampede GNU/Linux
www.suse.de	SuSE Linux
www.turbolinux.com	TurboLinux
www.yellowdoglinux.com	Yellow Dog Linux (PPC, RS-6000)

Beaucoup de ces distributions sont disponibles en différentes langues, mais toutes ne supportent pas l'ensemble des langues possibles : consultez le site web principal de chaque distribution pour plus de détails.

Revue rapide des distributions

Ce résumé n'est pas destiné à remplacer vos propres tests d'évaluation des diverses distributions de Linux, mais vous permettra d'aller dans la bonne direction. Toutes les versions citées exécutent un noyau de la série 2.2, fournissent les bureaux Gnome et KDE, des services de support réseau supplémentaires sur SAMBA et NFS, et utilisent Apache comme serveur web. Toutes les versions, sauf Corel et Mandrake, proposent une adresse e-mail

© copyright Éditions Eyrolles

pour leurs services de courrier électronique. Elles fournissent certains documents écrits, quand on achète le produit, ainsi qu'une documentation électronique (que l'on peut imprimer).

Important. Le numéro de version qui suit chaque nom de distribution est celui qui est en cours au moment de l'impression de cet ouvrage. La version suivante aura en principe un numéro plus élevé ou plus long, dans tous les cas.

Debian GNU/Linux 2.1

Debian est une distribution de Linux due à plus de 500 programmeurs bénévoles qui coopèrent *via* l'Internet. Debian bénéficie du soutien de VA Linux Systems Inc., O'Reilly et Associates Inc. et Silicon Graphic Inc. Le prix du package est de 20 dollars : il comprend un manuel séparé sur Debian GNU/Linux et le CD-ROM de la distribution. Si vous optez pour l'enregistrement de votre copie, vous pouvez recevoir gratuitement une copie de StarOffice. Le site web de Debian propose un accès illimité aux fichiers et à l'assistance, aux mises à jour gratuites permanentes, aux logiciels depuis leurs archives, et même des conseils gratuits !

Caldera OpenLinux 2.3

Caldera a une installation très habile qui s'inspire d'une architecture multitâche et on peut jouer à Tetris tandis que se déroule l'installation des paquetages. L'ensemble coûte 49 dollars et comprend un support technique. Caldera propose aussi beaucoup de suites bureautiques telles que WordPerfect, StarOffice, Koffice et Applixware. À l'origine, Caldera Linux dérive de Red Hat et partage beaucoup de ses caractéristiques.

TurboLinux 4.0

TurboLinux est un bon élève : un prix économique de moins de 20 dollars (ensemble à 49 dollars) et une édition serveur à 199 dollars qui comprend certains produits commerciaux, tel BRU, et un excellent service de support. C'est aussi une distribution dérivée de Red Hat qui a son propre jeu d'utilitaires d'installation et de gestion.

Linux Red Hat 6.1

Linux Red Hat 6.1 est la version préférée du marché, du moins d'après toutes les études sur Linux. L'installation graphique X est très soignée (voir certains détails déjà abordés) et comporte quelques utilitaires spécifiques de Red Hat comme DiskDruid et Kickstart, par exemple. Red Hat a contracté beaucoup d'alliances avec les grands noms des fabricants de PC et la plupart des éditeurs de bases de données renommés, comme Oracle et IBM. La version de

© copyright Éditions Eyrolles

base est à 29 dollars, mais la version Deluxe, qui propose une meilleure option d'assistance technique, est à 79 dollars. Red Hat annonce aussi régulièrement de nouveaux partenariats avec les plus grands constructeurs et éditeurs de logiciels ; aussi faut-il garder un oeil sur les articles de presse, si vous êtes intéressé par un éditeur ou un produit logiciel particulier.

Linux Mandrake PowerPack 6.1

Mandrake repose sur la distribution Red Hat, augmentée de certaines caractéristiques et de nombreux paquetages supplémentaires, pour un prix légèrement plus bas (55 dollars) que celui de la version Red Hat Deluxe (79 dollars). Mandrake comprend six CD-ROM, ce qui le rapproche de SuSE pour le nombre de CD-ROM, parmi les distributions citées ici.

SuSE Linux 6.3

SuSE Linux 6.3 est la version de Linux la plus connue en Europe : elle propose une multitude d'applications, de gestionnaires de bureaux et de logiciels de marque assortis, tout comme Mandrake. Le support technique est au-dessus de la moyenne et l'installation se fait aussi en mode graphique. De plus, elle contient une copie de VMWare ainsi que les suites bureautiques StarOffice et Applixware.

Récapitulation des serveurs de base de données

Voici, dans le tableau 1.3, une récapitulation orientée base de données, qui donne une rapide vue d'ensemble des principales versions serveur de Linux, avec le niveau de support technique de chacune d'elles ainsi que les serveurs de bases de données compris dans la distribution.

Tableau 1–3
Comparaison des distributions Linux sélectionnées

	Caldera	Debian	TurboLinux	Red Hat	Mandrake	SuSE
Mode d'installation	graphique	graphique	texte	graphique	texte	texte
Installation de système	Lizard	Programme d'installation Debian	Turbocfg	linuxconf	linuxconf	Yast2
Programme d'installation de paquetages	COAS, kpackage	Get-it, programme d'installation de Corel	Apt-get	RPM, GnoRPM	RPM, GnoRPM	RPM, GnoRPM

© copyright Éditions Eyrolles

Tableau 1–3
Comparaison des distributions Linux sélectionnées

	Caldera	Debian	TurboLinux	Red Hat	Mandrake	SuSE
Serveur de base de données	IBM DB2	Postgress SQL	IBM DB2	Mysql, Postgress SQL	Mysql	Mysql, Sybase, Informix
Support Technique	téléphone 90 jours, e-mail, web	Web	téléphone 60 jours, e-mail, web	téléphone 30 jours, 30/90/180 jours de service ftp et web (selon le prix)	téléphone 100 jours, web	téléphone 60 jours, e-mail, web

Les distributions suivantes ne figurent pas dans le résumé précédent, mais en voici brièvement les caractéristiques.

SPIRO-Linux

La version SPIRO-Linux est à 100 % compatible Red Hat et Mandrake, supporte 16 langues, exactement comme les autres, et fournit même le logiciel de synchronisation du Palm Pilot. Cette distribution vise les plates-formes Intel et Ultra de Sun et dispose de cinq configurations différentes d'installation en mode serveur, de trois configurations de station de travail, ainsi que d'options d'utilisation et de mise à jour. SPIRO comprend la suite Koffice et peut importer les documents Microsoft Word dans Linux. On trouvera plus d'informations à l'adresse www.spiro-linux.com.

Corel Linux

Corel Linux est plus qu'une version de station de travail : fortement orientée vers les nouveaux utilisateurs, elle met en œuvre un utilitaire d'installation graphique ultra-rapide. Corel Linux repose sur la distribution Debian et comprend une copie de WordPerfect 8 (il ne s'agit pas d'un portage complet de la suite bureautique disponible pour Win9x/NT). L'absence de support de courrier électronique, ou d'un serveur de base de données ne pose pas de problème car les connexions à tous les serveurs de base de données sont assurées. Vous pouvez encore installer PostgresSQL, compris dans la distribution Debian.

© copyright Éditions Eyrolles

Réfléchir avant la mise à jour ou la migration

Migrer une grande base d'un client utilisateur demande du temps et des ressources. Quelle que soit la manière dont on s'y prend, cela représentera un gros travail. Il faut prendre en compte l'état du réseau, du câblage et de la largeur de bande pour respecter les spécifications correctes de performance. La formation des utilisateurs ne sera nécessaire que dans le cas où les applications qu'ils ont l'habitude de manier sont d'un type totalement différent du serveur de base de données.

Si le produit de serveur de base de données ne change pas (par exemple, Oracle peut subsister sur une nouvelle plate-forme), les applications ne seront pas changées. S'il s'agit d'un nouveau service de serveur de base de données, comme dans le cas d'une migration de Microsoft Access ou Dbase vers Oracle ou Informix, il faudra envisager beaucoup de formations relatives aux nouvelles interfaces et applications associées.

S'il faut migrer une base de données existante vers ce nouveau service, il faudra former aux nouvelles interfaces les utilisateurs qui accèdent aux données déplacées. Le recyclage des utilisateurs à tout produit nouveau ou amélioré exigera des ressources ; et ce n'est qu'une partie de tout nouveau projet de développement ou d'amélioration. Les autres ressources à considérer sont : l'administration de base de données elle-même, les outils de développement destinés aux utilisateurs, le développement d'interfaces, les contraintes de largeur de bande du réseau additionnel, l'assistance et les services d'aide en ligne.

Compatibilité matérielle

La première considération concerne la compatibilité matérielle ; n'hésitez pas à prendre la disponibilité et le temps nécessaires à la recherche des solutions de compatibilité. En général, on peut vérifier les informations de compatibilité matérielle sur le site web de chacune des distributions (voir tableau 1.2) : souvent, le plus ancien est le meilleur. Le matériel vraiment obsolète peut toujours servir pour des usages généraux, tels que l'e-mail, le routage, les pare-feu, les services de fichiers et d'impression. Il faut mettre en perspective les solutions à base de recyclage de matériel dépassé face au coût relativement modeste des modèles récents, qui répondent souvent bien au besoin de faire tourner Linux. Linux a une courte expérience en tant que système d'exploitation et se heurtera à plus de difficultés en matière d'optimisation des matériels anciens.

© copyright Éditions Eyrolles

Bon marché, vous avez dit bon marché ?

Il était une fois un système : acheté, en version de test, voici un an, bien placé en tête de peloton, il subit une panne totale de l'alimentation électrique, malgré un an et un mois d'assistance continue — quoique légère. La garantie était d'un an. La pièce de remplacement est vendue au détail 30 dollars, mais les pertes occasionnées par un arrêt du système pendant une journée sont sans rapport avec l'économie faite en utilisant une alimentation de bas de gamme.

À titre de comparaison, les anciennes machines de 200 MHz, en service continu depuis plus de deux ans, n'ont pas encore eu de panne d'alimentation, ni la plupart de stations de travail à 100 MHz achetées il y a cinq ans. Vous avez ce que vous payez et il y a une raison pour laquelle les grandes compagnies commerciales vendent des matériels dédiés aux serveurs — c'est qu'ils sont conçus pour durer plus longtemps. C'est une niche du marché : moins on en construit, plus le prix est élevé. Il faut donc se préparer à payer le prix pour des produits serveur de qualité.

Dans la plupart des cas, votre serveur de base de données assurera la distribution de données critiques. Avoir le meilleur matériel possible sans s'ennuyer à rechercher les pilotes pour les matériels de conception très récente ? Avoir aussi du matériel éprouvé et fiable, apte à supporter la rigueur d'un service 24 heures sur 24, 7 jours sur 7 ? Les PC courants, vendus moins de 1000 dollars, ne disposent pas des meilleurs composants de haute fiabilité et ne sont pas conçus pour fonctionner en permanence plus d'un an ou deux : autant dire qu'ils sont conçus pour durer un minimum de temps et être remplacés souvent[2].

Aujourd'hui, la plupart des constructeurs de serveurs de marque (IBM, Compaq, Dell, HP et d'autres) proposent, sur au moins une de leurs lignes de produits, un Linux préchargé. Plusieurs constructeurs se spécialisent dans les serveurs fondés sur Linux (VA Linux Systems, Cobalt Networks et Penguin Computing, notamment). L'une ou l'autre de ces approches peut vous épargner beaucoup de temps et d'efforts en vous offrant un système Linux bien configuré et opérationnel.

La principale préoccupation sera de vous assurer que vous avez assez de puissance de traitement et de RAM. Le mieux est un serveur de base de données à processeurs multiples, deux fois la RAM que vous estimez nécessaire, équipé de la carte réseau la plus rapide possible avec une connexion directe au plus grand commutateur réseau disponible, et quatre fois l'espace disque dont vous avez besoin sur des contrôleurs SCSI multiples !

2. Les composants sont peu onéreux, en général !

© copyright Éditions Eyrolles

Environnement de test

Prévoir un environnement de test constitué d'au moins une machine serveur et de quelques machines client. Avoir aussi quelques utilisateurs reconnus qui testent les premières améliorations ; ayez un plan d'action pour la formation, le déploiement et le dépannage, et respectez-le. On fait souvent l'impasse de l'environnement de test pour respecter les délais. Un planning et une implémentation corrects se servent d'un environnement de test qui aplanit les difficultés pour tous ceux qui sont concernés.

Mise à jour

On trouve une option de mise à jour dans la plupart des installations Linux. Généralement, elle entraîne la réinstallation des versions les plus récentes de tous les produits installés, y compris les fichiers du noyau. Toutes les dépendances sont aussi installées, si nécessaire. Certaines distributions de Linux gardent la trace des paquetages installés et de l'endroit où les fichiers correspondants sont installés. Cette base de données est interrogée pour tous les paquetages présents : tous les nouveaux paquetages rapportés écraseront les anciens, ainsi que tous les nouveaux paquetages installés du fait des dépendances.

Important. Si l'on a déjà une version de Linux, il faut être attentif au fait que l'option d'installation d'une station de travail de certaines versions de Linux (Red Hat, par exemple) supprime toutes les partitions Linux (et ne laisse que les partitions des autres OS), puis crée et installe quelque chose de nouveau, alors que l'installation serveur sur certaines versions de Linux (là encore, Red Hat notamment) efface toutes les partitions sur tous les lecteurs et effectue une installation toute « neuve ».

Migrer depuis un autre système d'exploitation

Une grande question concerne la stratégie de migration à appliquer aux anciens fichiers d'information. On peut les laisser sur une partition du disque (FAT, FAT32, NTFS, etc.) et les mettre à disposition du système Linux. Il n'est pas nécessaire de supprimer l'ancien système d'exploitation ou un ancien système de fichier, s'il y a assez d'espace disponible pour Linux. On peut même garder l'ancien système intact, dans une configuration de double-amorçage (*dual boot*), jusqu'à ce que l'on soit prêt à achever la migration. Ce n'est pas la méthode recommandée si l'on a affaire à des centaines de stations de travail, car cela exige deux visites à chaque machine. On doit tester la transition, la planifier soigneusement, la dérouler, et prévoir quelques hoquets au démarrage.

© copyright Éditions Eyrolles

Toutefois, s'il ne s'agit vraiment que d'un serveur de base de données, il n'a pas du tout besoin d'un autre OS, ni de ses anciens fichiers, qui ne font qu'occuper de l'espace disque. Convertir les données et utiliser tout l'espace disque disponible sur la nouvelle machine de ce serveur. Garder une sauvegarde éventuellement, mais une fois la migration faite, on n'a plus besoin d'accéder aux anciennes données de toute façon.

Choisir une distribution

Le choix de la distribution est toujours délicat. Cela dépend du genre de locataire que vous souhaitez plutôt que de ceux que vous connaissez bien. Utilisez en premier celle qui semble le mieux répondre aux besoins, celle sur laquelle quelqu'un est déjà formé, ou celle que l'on peut télécharger gratuitement. Si vous êtes comme tout le monde, vous en chercherez probablement une qui propose de l'assistance gratuite, ou moyennant une légère redevance, comme Red Hat, Caldera, SuSE, Turbo Linux, ou une qui a le soutien d'une organisation commerciale telle que Compaq, IBM, VA Linux ou autres. La plupart des versions commerciales offrent une assistance. Il existe aussi des sociétés spécialisées dans l'assistance, indépendante des produits, comme LinuxCare.

Comment entamer le processus

Généralement, on entreprend de planifier les tâches, mettre les gens en place, tester soigneusement le plan, et puis de s'y tenir. Plaçons-nous dans le cas où vous avez déjà un projet, sur lequel vous travaillez, qui exige un serveur de base de données dédié. Comme on l'a vu auparavant, Linux est un excellent choix pour un serveur sous-jacent. Il faut inclure dans votre plan quelques tests du serveur de base de données, le plus tôt possible en phase de développement. Il ne faut pas faire l'achat du matériel à la dernière minute ou selon une décision inconsidérée — à vue. Les matériels changent tous les jours, et de nouvelles versions de Linux sortent tous les six mois. S'il s'agit vraiment d'un grand projet, allez vers des grands constructeurs et demandez-leur une démonstration de leur matériel fonctionnant sous Linux. Testez bien votre conception initiale avec eux, tout en gardant à l'esprit vos besoins à long terme.

© copyright Éditions Eyrolles

Formation de l'équipe

Il y a beaucoup de sociétés qui proposent des formations maison, telle Global Knowledge Inc. — voir www.globalknowledge.com. D'autres proposent les formations offertes par les distributions de Linux elles-mêmes. Sur le Web, la recherche sur le thème « Formation Linux » renvoie des listes d'organisations comme www.linuxtraining.org, www.learningtree.com ou bien www.linux-systemsgroup.com.

On ne peut pas faire l'impasse d'une formation correcte des utilisateurs et de l'équipe d'assistance et d'administration. Comprenez aussi : ce n'est pas parce qu'un cours s'intitule « Introduction à ... » (certains admettent des candidats sans aucune condition préalable), que vous pouvez y envoyer n'importe qui. Une grande société de formation professionnelle offre des parcours de très haut niveau — parfois même, le parcours d'introduction commence à un niveau assez avancé.

Solutions à présenter à votre directeur

- **Coût.** Aucun doute à ce propos, le coût nul de la licence du système d'exploitation est un avantage. Quand on calcule les coûts de licences pour les plates-formes en concurrence, s'assurer de bien prendre en compte le coût des licences pour les clients qui utilisent le serveur, et le coût ultérieur des mises à jour. Ils peuvent être très élevés pour certains systèmes d'exploitation. Mais la licence gratuite ne signifie pas qu'il n'y a aucune assistance ; il existe un support commercial. Les grands constructeurs d'ordinateurs se mettent à offrir des PC préconfigurés avec Linux, Red Hat la plupart du temps. Que des compagnies comme Compaq, Dell, et IBM aient décidé de supporter Linux sur leur matériel est un grand « plus » en faveur de l'utilisation de Linux.
- **Fiabilité.** Il n'y a jamais de raison de redémarrer Linux : les pilotes et les applications peuvent être rechargés à tout moment. La plupart des systèmes d'exploitation des serveurs Internet sont propulsés par Linux ou d'autres systèmes d'exploitation de type UNIX, parce qu'ils sont extrêmement dignes de confiance. UNIX domine le marché haut de gamme des serveurs de production et des stations de travail graphiques parce qu'il a une fiabilité et une faculté de reprise qui lui permettent de continuer à fonctionner dans n'importe quelles conditions.
- **Sécurité.** La force de Linux et d'Unix sur l'Internet réside dans leur modèle de sécurité, mûr et robuste. Le noyau est bien protégé de l'accès direct de la part d'un utilisateur quelconque, et le système de fichiers, conçu pour être sûr, est éprouvé depuis des années sur l'Internet. La plupart des brèches dans la sécurité ne sont pas des faiblesses qu'un utilisateur moyen, ou même au-dessus de la moyenne, sera capable de mettre à

© copyright Éditions Eyrolles

profit au niveau d'un système d'exploitation — elles viennent de pratiques peu soucieuses de la sécurité des utilisateurs. Windows 98 n'offre aucune sécurité réelle intégrée. La sécurité, dans Linux, est fondée sur le système de fichier, et opère au niveau de chaque fichier.

Faire une analyse correcte des coûts

Il faut encore faire une analyse particulière, liée aux coûts induits par l'ajout d'un autre système d'exploitation dans votre réseau : l'assistance nécessaire (formation de l'équipe — la formation des clients étant minimale), les problèmes d'intégration (le moins possible), les interfaces (différents types de fenêtres mais les mêmes concepts de base), et les économies engendrées.

Il faut disposer de ressources en test, formation, recyclage, et des ressources durables d'aide en ligne, quel que soit le système d'exploitation. Avec un système d'exploitation puissant, sûr et fiable tel que Linux, vous pouvez optimiser certaines de ces ressources.

Résumé

Le phénomène Linux est durable, car c'est un produit attrayant : c'est le sentiment à la base, renforcé par une facilité d'installation et d'intégration qui va en s'améliorant. C'est une alternative solide à tout système d'exploitation Windows dans la mesure où ce n'est PAS Windows en soi. Linux est chez lui dans le domaine des serveurs comme dans celui de l'environnement de bureau. Gardez à l'esprit l'idée que Linux est une version d'UNIX, et qu'il faudra quelqu'un d'expérimenté sur Linux ou sur UNIX pour le gérer. L'interface est similaire à celle de Windows, sans être exactement la même, et il n'y a rien de ce que l'on peut faire avec Windows qui ne puisse se faire dans Linux. Il y a une abondance d'applications utilisateur disponibles sur Linux qui remplissent les mêmes fonctions, quoique différemment parfois. Les avantages principaux sont le coût, la fiabilité et la sécurité que Linux offre en tant que système d'exploitation, sans oublier sa présence croissante dans la cour des grands tels que Oracle, Informix, IBM, Sun, Dell, Compaq et beaucoup d'autres.

© copyright Éditions Eyrolles

FAQ

Q. Vais-je perdre l'accès à Windows ?

R. Oui et non. Oui, si vous le voulez ; mais vous n'y êtes pas obligé. Cela sera plus évident dans la section relative à l'installation. À la base, on peut ajouter Linux comme système d'exploitation secondaire sur la plupart des PC, sans perdre la configuration existante, si l'on dispose de suffisamment d'espace disque. Vous pouvez aussi décider d'enlever toute trace d'autres systèmes d'exploitation si vous le souhaitez et utiliser la machine avec Linux seulement. En réalité, il vaut mieux éliminer tout autre système d'exploitation sur un serveur de base de données. Ne consommez pas votre espace disque en ayant plusieurs systèmes quand vous pouvez n'en utiliser qu'un seul à un moment donné (sauf si vous utilisez VMWare — sachant que VMWare est plus utile sur une station de travail que sur un serveur, parce qu'il surcharge la machine en exécutant deux systèmes d'exploitation à la fois).

Q. Est-ce que j'ai besoin de garder Windows ?

R. Oui, si vous ne pouvez absolument pas vivre sans lui. Non, si vous voulez migrer ou si cette machine est destinée au développement. Comme tout système d'exploitation, vous pouvez toujours le réinstaller ultérieurement. En général, pour un serveur de base de données, vous n'en utiliserez qu'un seul.

Q. Comment installer un dual boot (ou triple, quadruple, etc.) avec mon système courant d'exploitation, ou mes systèmes d'exploitation ?

R. Se souvenir que, sauf s'il s'agit d'un système de développement, il n'y a ni nécessité, ni intérêt à conserver deux systèmes d'exploitation sur une machine qui est vouée à être un serveur. Cependant, lors de l'installation, vous pouvez demander au système d'ajouter la fonctionnalité de dual boot à l'aide de **lilo**, ou bien ajouter le boot Linux à l'un des gestionnaires de boot courants sur les PC à base d'Intel.

Q. Linux peut-il exécuter Microsoft Office 97 ?

R. Pas vraiment, car il n'est pas natif. En se servant d'un émulateur, tel que WINE, on peut exécuter tout ou partie d'Office 97. On peut aussi utiliser l'une des nombreuses suites similaires, mais *pas exactement* identiques, telles StarOffice ou WordPerfect. Chacune d'elles peut lire et écrire le format Office 97 de la plupart des documents.

Q. Quid de l'assistance ?

R. Pour l'assistance, démarrer en pensant à celle dont vous avez besoin dans l'immédiat. Faites comme vous avez fait quand vous avez démarré Windows 9x ou NT. Donnez à quelqu'un une chance de briller, avec une

© copyright Éditions Eyrolles

promotion à la clé, ou demandez un volontaire. Vous pouvez certainement trouver quelqu'un qui connaît bien le sujet, joue déjà avec à la maison ou l'a utilisé à l'école. L'assistance est disponible gratuitement *via* les services de news et les groupes d'utilisateurs sur l'Internet, et un support commercial payant est proposé par beaucoup d'organismes. Voir la liste des éditeurs au tableau 1.3.

Q. Comment désinstaller Linux ?

R. Techniquement, il suffit de repartitionner le disque, de rassembler les nouvelles partitions actives en une seule, de transférer quelque OS de type DOS sur cette partition active, puis de redémarrer. Voilà, Linux est parti pour toujours. On peut faire cela avec tout système d'exploitation sur PC : on obtient le même résultat.

Si vous avez Windows 9x, utilisez une disquette de démarrage de Windows 9x, et exécutez la commande :

```
A :> FDISK /MBR
```

qui éliminera l'amorce **lilo** et redémarrera automatiquement Windows 9x. Repartitionner ou reformater toutes les anciennes partitions Linux pour leur rendre le format FAT.

Si vous avez Windows NT, sur une autre partition, il est un peu plus difficile de reconstruire le MBR NT, mais vous pouvez le faire avec l'option de secours en forçant NT à réécrire le bloc MBR.

Q. Quelles sont les applications qui fonctionnent sous Linux ?

R. On trouve beaucoup d'applications un peu partout sur l'Internet. Du fait qu'il s'agit de distributions Linux, les applications doivent être spécifiquement orientées Linux, tout comme les applications Windows 9x/NT doivent s'exécuter sur leurs propres plates-formes. Il existe plusieurs milliers de produits pour Linux dont on ne trouve pas trace dans les magazines PC (orientés Windows), la plupart du temps parce qu'ils sont gratuits, et qu'aucune publicité n'y est attachée. Vous pouvez les trouver sur l'Internet, ou prendre connaissance des versions commerciales, annoncées dans votre magazine favori ou dans votre moteur de recherche web. Ayez conscience que tous les grands services de serveur fonctionnent sur Linux : gestionnaires de base de données, gestion de réseaux, pare-feu, services web, services d'e-mail, et ainsi de suite.

Q. Je n'ai jamais installé de système d'exploitation, est-ce difficile ?

R. Tout est difficile en l'absence des connaissances ou des outils requis. Linux est convivial, autant que peut l'être une installation, mais il s'efforce d'être le moins dirigiste possible. Mettez le CD-ROM dans votre machine et allumez-la. Les machines les plus récentes démarreront

© copyright Éditions Eyrolles

l'installation automatiquement, tandis que d'autres doivent être démarrées manuellement. Même dans ce cas, ce n'est pas trop difficile. Voir le chapitre 2, qui couvre l'installation de Red Hat 6.1 en détail.

Q. Est-ce que je dois m'y connaître en UNIX pour utiliser Linux ?

R. Oui, j'en ai bien peur. Bien que le monde graphique des fenêtres soit agréable, ce n'est qu'une des méthodes de gestion. Et c'est la gestion qui fait la différence entre les versions de systèmes d'exploitation. Il faut avoir une idée claire de la manière dont UNIX/Linux fonctionnent sur des tâches précises avant de se colleter à un problème. Une fois que vous connaissez les bases, vous pouvez vous mettre en chasse des problèmes les plus coriaces.

© copyright Éditions Eyrolles

2

Installation de base de Red Hat Linux

Dans ce chapitre

- Introduction
- Préliminaires
- Installer Red Hat
- Installation personnalisée
- Autres versions de Linux

© copyright Éditions Eyrolles

Introduction

Ce chapitre aborde les concepts de base qu'il faut connaître pour installer une distribution Linux, et présente des exemples spécifiques d'une installation Red Hat. Les distributions Linux fournissent toujours une pléthore d'utilitaires et de produits, parmi lesquels il faudra faire un choix. Beaucoup comprennent un exemple, ou une version libre d'un serveur de base de données.

Linux est aussi un serveur de nombreux services de réseau et de production ; c'est pourquoi il faudra prendre des options, en préalable à toute installation. Techniquement, on peut exécuter toutes les combinaisons de services suivants : protocole d'adressage IP dynamique (DHCP), serveur de nom de domaine (DNS), ftp, un produit de base de données comme Oracle (non fourni dans la plupart des distributions), ou le serveur PostgreSQL (version libre fournie avec certaines versions de Linux). On fait appel à tous ces services depuis une seule machine physique, ou bien à partir de toutes sortes de combinaisons de machines.

Important. Les copies d'écran utilisées dans ce chapitre ont été obtenues lors d'une installation de Linux sur une machine hôte NT 4.0 fonctionnant avec VMWare. Elles sont données à titre d'illustration ; les barres de titre, telles que celles des figures 2.2, 2.3 et suivantes, n'apparaissent pas pendant l'installation.

Machine physique séparée

Evidemment, il est hautement recommandé de dédier un serveur séparé à la gestion de la base de données. Cela procure un certain confort en matière de réactivité, de sécurité et de fiabilité. Les applications les plus courantes fondées sur les bases de données sont de conception complexe et consomment beaucoup de ressources, tant processeurs que disques. Il ne faudrait tout de même pas que votre serveur de base de données se retrouve en concurrence, pour les cycles CPU ou les accès disque, avec d'autres services (web, ftp, fichier ou impression, par exemple) ! Cela réduirait grandement la performance de tous les services en général.

En traitant à part la gestion des bases de données, vous protégez aussi mieux le système des intrusions, et maintenez les données, ainsi que leur sauvegarde, totalement séparées, à l'abri des autres processus. Les principaux types d'attaques contre le système trouvent la plupart du temps une voie dans les services standard du réseau. Si vous maintenez ces services sur une autre machine, les intrus ne seront pas en mesure de s'attaquer à votre serveur de base de données. Même les mots de passe de l'administrateur seront différents entre ces systèmes pour plus de sécurité contre les pirates qui feraient irruption sur d'autres serveurs de votre réseau.

© copyright Éditions Eyrolles

Choisir une distribution Linux

Il faut ensuite prendre en considération la version de distribution de Linux. Il y a beaucoup de choix ; toutes supportent, sans doute, une base de données certifiée sur Linux, jusqu'à un certain point ; mais les versions commerciales les plus courantes auront probablement déjà un service de support ou des accords avec l'éditeur du logiciel de base de données. Red Hat a un support pour beaucoup de logiciels de base de données, tel Oracle, et d'autres produits issus de constructeurs comme Dell, Compaq ou IBM, pour n'en citer que quelques-uns.

Décisions préliminaires à l'installation

Un projet doit être conçu étape par étape. Dans cette section, nous abordons certains sujets de base qu'il faut prévoir de traiter avant de commencer l'installation. Cela comprend les niveaux de performance souhaités, maintenant et à l'avenir, et influe sur les décisions en matière de matériel et de logiciel, avant toute installation.

La partie la plus critique de l'approche du serveur concerne les divers aspects matériels, esquissés dans les prochaines sections. Cela reste vrai quel que soit le système d'exploitation choisi.

Matériel

Bien que cela puisse sembler bizarre en cette période d'ordinateurs domestiques à bas prix, les éditeurs de serveur de renom sont dits grands justement parce que leurs serveurs haut de gamme sont conçus pour fonctionner en permanence, avec des rattrapages de panne, une maintenance et un accès aux services aisés. C'est là qu'il faut prendre son temps pour trouver une machine qui puisse gérer de longues périodes de fort débit d'I/O, exécuter plusieurs processus simultanément, transférer beaucoup de données entre les disques et la mémoire, et de la mémoire au processeurs, conserver en mémoire la plupart des processus et disposer, sous une forme ou une autre, de possibilités de rattrapage matériel - RAID - en cas d'erreur ou de panne. Assurez-vous que votre nouveau système peut admettre facilement des extensions dans les zones de contrôleurs de disque, les baies de lecteurs de disque et la RAM.

© copyright Éditions Eyrolles

Architecture multiprocesseurs

Les serveurs de base de données consomment beaucoup de ressources CPU. Les processeurs à bon marché peuvent fonctionner en parallèle pour fournir un débit étonnant et des niveaux de charge maximale. Bien que l'on n'obtienne pas 100 % d'amélioration à chaque addition de processeurs, le bénéfice obtenu, en gardant tout en mémoire locale, l'emporte sur le coût qui résulte d'un serveur multiprocesseurs.

RAM

Le facteur le plus critique, après le choix du processeur (ou des processeur), est la taille de la RAM : il en faut beaucoup. Faites une analyse préliminaire des besoins de la base de données envisagée, et ajoutez une marge raisonnable. La taille minimale raisonnable de RAM pour démarrer est approximativement de 256 Mo pour les bases de données les plus courantes.

Disques

Le sous-système disque vient immédiatement après, dans l'ordre des caractéristiques importantes. Les contrôleurs multiples fournissent un meilleur débit et un système RAID (*Redondant Array of Independent Disk*) de niveau 5, une bonne gestion des agrégats de partitions avec parité, et donnent même de meilleures performances dans la plupart des situations - mais pas toutes. Dans certains cas, spécialement avec les bases de données, il vaut mieux opter pour un système RAID de niveau 0, avec des agrégats de partitions sans parité, ou bien de niveau 1, avec mise en miroir de disques. Gardons à l'esprit le coût associé à chaque niveau RAID, que nous verrons mieux dans la prochaine section consacrée au RAID.

Dans l'absolu, on n'a pas besoin de RAID : des disques ultra-rapides de bonne qualité, dotés d'un bon nombre de caches, feront l'affaire. Si vous avez une sauvegarde et que le temps d'arrêt nécessaire à la restauration ne vous coûte pas trop, utilisez ce que vous voulez. Souvenez-vous seulement que, si votre système de saisie de commandes se situe sur le serveur et qu'il tombe en panne, tout un chacun devra patienter jusqu'à ce qu'il fonctionne à nouveau — *chacun*, c'est-à-dire votre équipe et vos clients. Avez-vous vraiment les moyens d'assumer ? A comparer le faible coût du matériel et les pertes potentielles de revenu, on a tendance à opter pour l'un des formats de RAID facilement récupérable, même s'il n'assure pas une disponibilité 24 heures sur 24.

© copyright Éditions Eyrolles

RAID

Le niveau RAID 1, avec mise en miroir, est une bonne option pour les disques système, où réside le système d'exploitation. Les fichiers de base de données doivent être sur des disques entièrement séparés du système d'exploitation : un matériel RAID 5 donnera les meilleurs résultats, en matière de débit et de fiabilité.

Coût des systèmes RAID

Le niveau RAID 1 est le plus cher. Dans ce cas, si l'on utilise deux disques de la même taille, on ne dispose que de l'espace de l'un d'eux pour le système ; l'autre est une duplication, et l'on utilise 50 % des deux disques. Si l'on utilise trois disques pour la mise en miroir, le taux effectif est de 33 % car un seul disque est réellement disponible — en substance, les deux autres ne sont que la duplication du premier.

Avec le niveau RAID 0, on utilise 100 % des disques et l'on obtient habituellement une amélioration de la vitesse de lecture et d'écriture : tout l'espace disque est disponible, en revanche il n'y a *aucune* restauration. Si l'un des disques subit une panne, ils sont tous perdus : il faut alors tous les reconstruire et restaurer toutes les données de tous les disques. Cela n'est *pas recommandé* pour les données critiques, mais peut être utilisé pour des données provisoires ou des résultats intermédiaires. Il existe une alternative acceptable : un ensemble de services RAID matériels qui gère, à la base, les disques multiples. On peut alors utiliser le logiciel RAID 0 pour les combiner en un grand disque unique, ce qui facilite les accès serveur.

Le niveau RAID 5 se sert en réalité de l'espace de l'un des disques (en effectuant une permutation de tous les disques du groupe RAID) pour produire un bloc de parité, utilisé pour regénérer les données manquantes après la perte d'un disque entier. Comme il ne peut y avoir qu'un seul disque en panne à un instant donné, il faut s'efforcer de le récupérer avant qu'un autre ne tombe en panne ; mais, bien entendu, il y a des exceptions. Le minimum est de trois disques, le maximum de trente-deux, selon le système d'exploitation utilisé. Avec trois disques, on n'utilise que deux disques sur trois, et l'on a un taux d'utilisation de 67 % ; avec quatre disques, trois sur quatre pour 75 % ; quatre sur cinq pour 80 %, et ainsi de suite. La meilleure utilisation est de 31 sur 32 pour un taux d'utilisation d'environ 96 % avec récupération automatique. C'est la solution recommandée pour les données particulièrement critiques.

RAID matériel ou logiciel

Linux fonctionne pour les niveaux RAID 1 et 5, en version logicielle. C'est une excellente option, si l'on ne retient pas la solution matérielle. Elle est

© copyright Éditions Eyrolles

implémentée avec un minimum de deux ou trois disques, respectivement, et avec le service du démon **mdrecoverd.** On a le choix entre des disques entiers, ou simplement des partitions de même taille, pour implémenter les ensembles RAID. L'installation est clairement décrite dans la documentation HOWTO-RAID.

Interface(s) réseau

Il y aura un autre goulot d'étranglement dans votre système. Plus le canal est gros (débit des données), mieux il fonctionne. La partie la plus lente de tout système est le système de disques. Il fonctionne en millisecondes de temps d'accès moyen, la mémoire du système en nanosecondes, et le réseau se situe au milieu, en général en micro-secondes. Prenez des cartes de 100 Megabits et un commutateur (*switch*) à la place d'un noeud, ou bien deux cartes d'interface réseau (NIC) avec deux noeuds séparés, si possible, et répartissez la charge. Après le système de disque, c'est le second goulot d'étranglement qui gêne le flux de données entre votre serveur et vos clients.

Mesures de sauvegarde

Le plus grand lecteur de bande devient plus grand tous les mois. Une bonne sauvegarde, parfaitement efficace, mais néanmoins monobande, peut conserver un maximum d'environ 80 Go. Choisir au moins 2 Go, pour tenir compte de la redondance et de la vitesse. Il y a beaucoup d'autres options que les bandes ; par exemple, les disques Iomega Jaz, les CD-ROM, les CD réinscriptibles, les DVD-ROM, qui peuvent offrir une meilleure disponibilité que les bandes, à un coût à peine plus élevé. On peut aussi opter pour les systèmes de gestion de disques les plus récents, qui proposent la migration des données depuis les disques RAM high-need, vers des lecteurs de disque medium-need, des disques Jaz ou des CD-ROM low-need, ou sur bande pour des archives rarement utilisées ; tous ces moyens constituent un seul grand système, considéré par votre machine comme un seul grand disque. C'est vraiment un excellent produit de mise à disposition de vos données sur le serveur. Les bandes sont le moyen traditionnel, mais on peut acheter deux disques Jaz, ou un autre lecteur de disque de 6–10 Go (capacité qui ne cesse d'augmenter), qui auront un temps de restauration beaucoup plus rapide. Installer ces disques sur un autre serveur et les utiliser comme média de sauvegarde — rapide à sauvegarder, rapide à récupérer, et moins cher. Si l'autre serveur est hors site, cela signifie que vous avez intégré une restauration désastreuse, compte tenu de la distance. Vérifiez toutes vos options avant de vous lancer, en ayant à l'esprit que le temps de restauration est un élément majeur de votre choix.

© copyright Éditions Eyrolles

Assistance

Tous les grands constructeurs, ainsi que beaucoup d'éditeurs de Linux, ont fait des services d'assistance leur cheval de bataille, visant des clients tels que votre société. Vous payez un bonus pour un produit en prime, sachant que, la plupart du temps, vous êtes gagnant.

Remarque. Les services d'assistance varient selon les versions de Linux :

- Si vous avez acheté les produits Deluxe ou Professional officiels, Red Hat offre 180 jours de support ftp libre, 90 jours de support sur le web, et 30 jours de support téléphonique à l'installation.
- Debian GNU/Linux fournit un accès non limité à ses sites de support ftp et web.
- Les acheteurs de TurboLinux obtiennent 60 jours de support technique.

Préliminaires à l'installation

Pour une machine de développement, il suffit d'ajouter quelques disques à une machine existante, puis d'installer Linux sur ces disques, de placer le système en double amorce, et de redémarrer l'OS de son choix. C'est un scénario courant pour beaucoup d'administrateurs qui veulent l'essayer d'abord, le vérifier avec un chargement simulé, puis le déployer sur le terrain.

Pour un serveur réel, à plein temps, dédié à des tâches critiques, il faut dédier la machine au service de la base de données, et à rien d'autre de préférence.

Installer Red Hat

Linux Red Hat est la distribution la plus complète ; elle est unique. Cependant, quelques autres distributions importantes en dérivent, à des degrés divers. Red Hat a été l'une des premières versions commerciales de la distribution Linux à offrir de l'assistance et à ajouter des caractéristiques conviviales : c'est pourquoi nous le présentons ici.

Un parcours d'installation rapide

Insérer le CD-ROM marqué #1 dans un lecteur de CD-ROM « bootable » et redémarrer la machine. Le système va lancer sa séquence normale de démarrage qui comprend l'identification des périphériques matériels, le test de la mémoire, et la vérification de base des systèmes. À ce point, la disquette, ou

© copyright Éditions Eyrolles

le CD-ROM, commencera le processus de boot. Si le système ne détecte pas le CD-ROM et démarre autrement, il faut modifier les paramètres du BIOS pour ajouter le CD-ROM au début de la séquence de boot ; sur certaines machines anciennes, il faudra utiliser une disquette de boot pour démarrer le processus.

Après que les fichiers initiaux ont été chargés en mémoire, s'affiche un écran d'accueil de Red Hat contenant quatre options (non figurées ici). Appuyer sur la touche **ENTER** pour continuer l'installation. On peut aussi taper **TEXT** pour sélectionner une installation en mode texte, mais elle n'est pas traitée ici spécifiquement ; bien que les questions posées soient identiques, l'interface est différente. La figure 2.1 montre la sortie type du démarrage initial d'un noyau Linux.

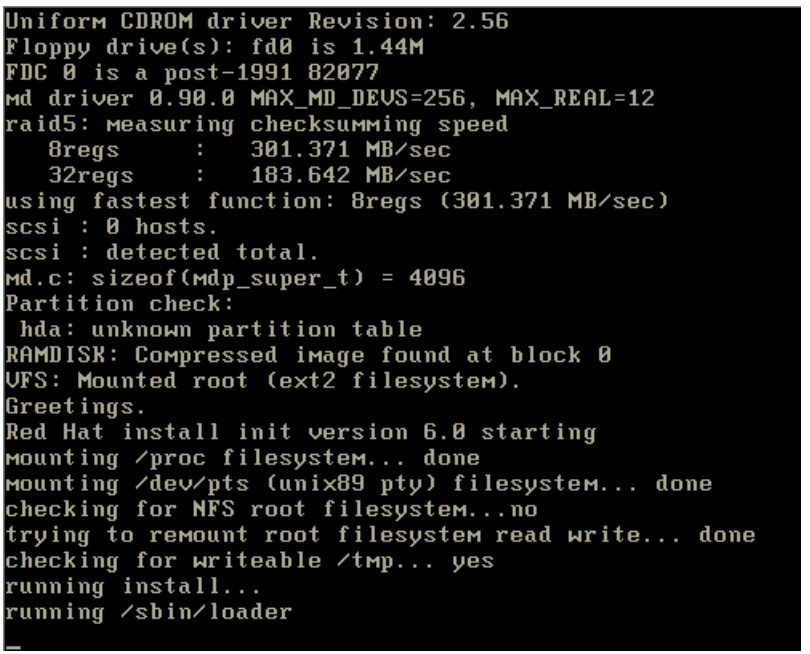
Uniform CDROM driver Revision: 2.56
Floppy drive(s): fd0 is 1.44M
FDC 0 is a post-1991 82077
md driver 0.90.0 MAX_MD_DEVS=256, MAX_REAL=12
raid5: measuring checksumming speed
8regs : 301.371 MB/sec
32regs : 183.642 MB/sec
using fastest function: 8regs (301.371 MB/sec)
scsi : 0 hosts.
scsi : detected total.
md.c: sizeof(mdp_super_t) = 4096
Partition check:
hda: unknown partition table
RAMDISK: Compressed image found at block 0
VFS: Mounted root (ext2 filesystem).
Greetings.
Red Hat install init version 6.0 starting
mounting /proc filesystem... done
mounting /dev/pts (unix89 pty) filesystem... done
checking for NFS root filesystem...no
trying to remount root filesystem read write... done
checking for writeable /tmp... yes
running install...
running /sbin/loader

Figure 2–1
Démarrage typique de Linux (après l'écran de démarrage préliminaire)

L'installation teste la compatibilité du matériel avec X Window, puis démarre une installation en mode graphique interactif de type X. L'installation graphique affiche d'abord le nom et le logo de Red Hat avant de se poursuivre.

La première étape de l'installation est la sélection de la langue (figure 2.2). Remarquer que Linux peut être installé dans de nombreuses langues. Notons aussi que le CD-ROM fourni aux États-Unis ne contient pas d'autre langue

© copyright Éditions Eyrolles

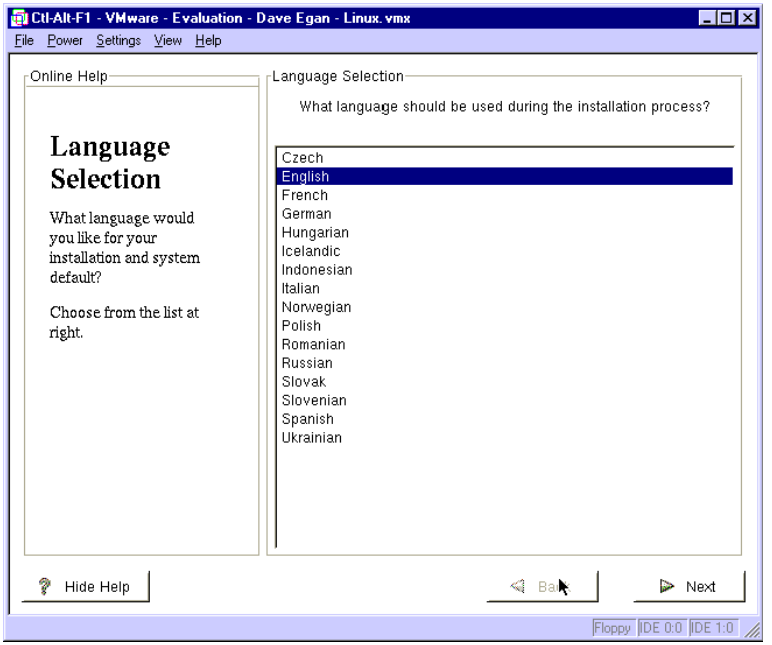

Figure 2–2
Choix de la langue

d'installation ; télécharger les fichiers sur le site web/ftp de Red Hat, ou commander la version CD-ROM spécifique de la langue. L'écran suivant combine en un seul écran ce qui apparaît en trois écrans séparés en mode texte, et permet de choisir les options relatives au clavier et à la souris. Remarquez que Linux a déjà détecté le matériel de base de votre système (figure 2.3) ; ce sera le cas dans beaucoup d'écrans.

La figure 2.4 présente les options de sélection de la souris. Linux a une préférence pour une souris à trois boutons. Cependant, vous avez probablement une souris à deux boutons : changez la sélection pour une souris à deux boutons du type adapté, soit le type bus (connue aussi comme PS/2 — celle qui a un connecteur « aussi petit qu'un stylo »), soit le type série (connecteur à neuf broches).

Important. Il faut indiquer explicitement les bonnes options pour la souris car le système X Window préfère trois boutons. On simule le bouton du milieu en pressant ensemble les deux autres boutons, en même temps.

L'écran fait encore apparaître le logo Red Hat. Il indique que vous êtes sur le point d'installer le logiciel et de faire les choix relatifs aux partitions du disque, si nécessaire. Vous avez la possibilité à ce point de revenir en arrière et de faire des changements, le cas échéant.

© copyright Éditions Eyrolles

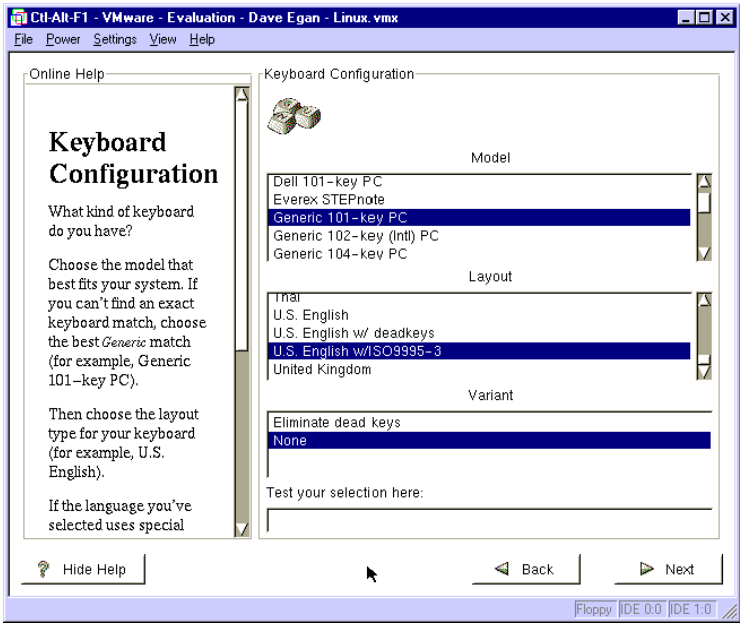

Figure 2–3
Sélection de clavier

Choisir un type d'installation

Le type d'installation (figure 2.5) est *très important*. Il vous est demandé de prendre une grande décision sur l'installation de votre Linux. Avec Red Hat, vous avez trois options de base : serveur (**Server**), personnalisée (**Custom**), et deux versions de la station de travail avec, au choix, le bureau Gnome ou le bureau KDE. Une quatrième option, mise à jour (**Upgrade**), n'est à choisir que si vous avez déjà une précédente version de la même distribution Linux sur la machine. On ne peut mettre à jour qu'une version précédente du même Linux, Red Hat dans notre cas.

Installation d'une station de travail

L'option « station de travail » est suffisante dans le cas d'un client ou d'une machine de test, mais elle n'est pas recommandée pour un serveur réel. Elle installe un Linux de base avec une interface X Window, soit Gnome soit KDE. Elle est prédéfinie pour répartir l'espace disque disponible en trois partitions et tout installer dans la plus grande des trois. Si l'on envisage d'ajouter des disques par la suite, on peut, d'un point de vue technique, faire ce choix. Si l'on a beaucoup d'espace disque, on peut ainsi créer une machine

© copyright Éditions Eyrolles

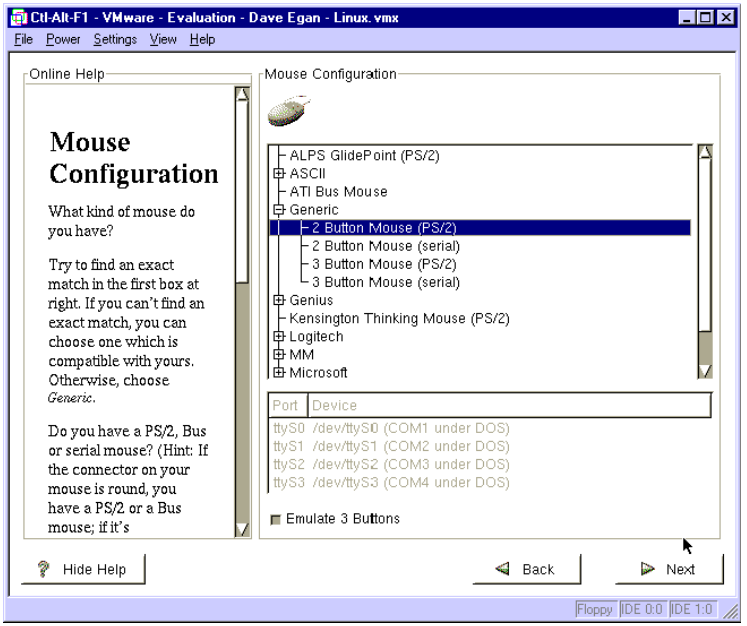
Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
Mouse Configuration
What kind of mouse do you have?
Try to find an exact match in the first box at right. If you can't find an exact match, you can choose one which is compatible with yours. Otherwise, choose Generic.
Do you have a PS/2, Bus or serial mouse? (Hint: If the connector on your mouse is round, you have a PS/2 or a Bus mouse; if it's
Mouse Configuration
ALPS GlidePoint (PS/2)
ASCII
ATI Bus Mouse
Generic
2 Button Mouse (PS/2)
2 Button Mouse (serial)
3 Button Mouse (PS/2)
3 Button Mouse (serial)
Genius
Kensington Thinking Mouse (PS/2)
Logitech
MM
Microsoft
Port
Device
ttyS0 /dev/ttyS0 (COM1 under DOS)
ttyS1 /dev/ttyS1 (COM2 under DOS)
ttyS2 /dev/ttyS2 (COM3 under DOS)
ttyS3 /dev/ttyS3 (COM4 under DOS)
Emulate 3 Buttons
Hide Help
Back
Next
Floppy
IDE 0:0
IDE 1:0

Figure 2–4
Sélection de la souris : à deux boutons ; les sélections automatiques émulent une souris à trois boutons

de développement et installer la base de données, dans cette grande partition ou sur un autre disque, au choix. Par défaut, s'il y avait un autre système d'exploitation sur cette machine, il sera automatiquement conservé et proposé dans les options de l'amorce (ce n'est pas affiché, car c'est à vous de savoir qu'il est là). Ce n'est pas le type d'installation recommandé pour un serveur.

Installation d'un serveur

L'installation d'un serveur est un autre choix possible, car cela fait un bien meilleur travail de partitionnement du disque, dans une conception plus orientée gestion, ce qui est souhaitable. Elle installe aussi un ensemble de logiciels serveur, mais *n'installe pas* le système X Window par défaut. Cette installation supprime encore toutes les partitions du système d'exploitation précédent, quel que soit leur type, crée six partitions, de taille minime pour la plupart, puis installe tous les paquetages serveur et quelques paquetages utilisateur. C'est la meilleure manière de démarrer un serveur Linux, qui offre tous les services réseau prêts à fonctionner ; c'est aussi la plus facile sur une nouvelle machine. Assurez-vous que vous disposez de suffisamment d'espace pour votre application de base de données et ses données.

© copyright Éditions Eyrolles

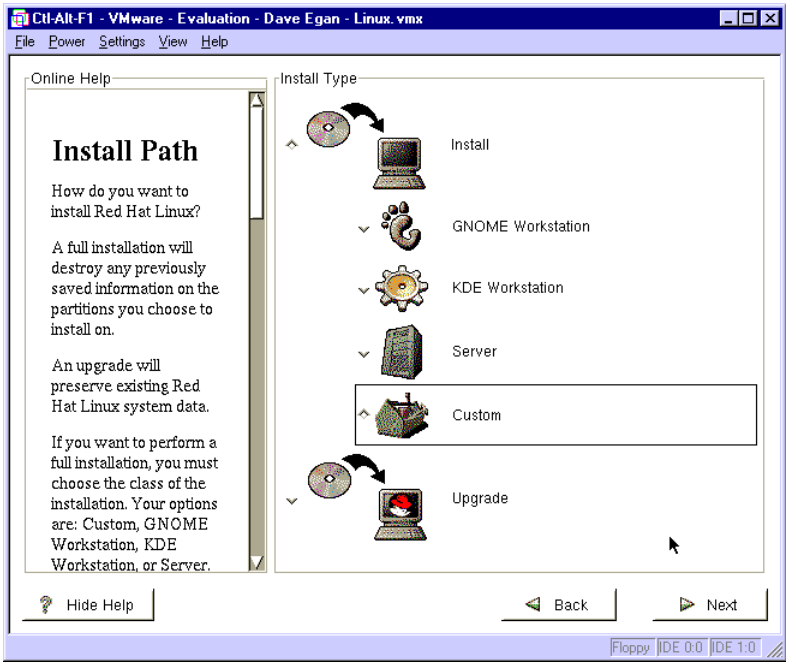
Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
Install Path
How do you want to install Red Hat Linux?
A full installation will destroy any previously saved information on the partitions you choose to install on.
An upgrade will preserve existing Red Hat Linux system data.
If you want to perform a full installation, you must choose the class of the installation. Your options are: Custom, GNOME Workstation, KDE Workstation, or Server.
Install Type
Install
GNOME Workstation
KDE Workstation
Server
Custom
Upgrade
Hide Help
Back
Next
Floppy
IDE 0:0
IDE 1:0

Figure 2–5
Type d'installation

Mise à niveau

La mise à jour n'est évidemment prévue que pour un système qui contient déjà une version de Red Hat, et ne s'applique à aucune autre version de Linux. La mise à jour ne remplace que les logiciels déjà installés par les versions plus récentes et installe tout nouveau groupe de logiciel requis par les dépendances.

Installation personnalisée

L'installation personnalisée est le meilleur choix pour quelqu'un qui sait exactement ce qu'il veut. Vous pouvez implémenter une conception spécifique pour vos disques — par exemple définir la taille des partitions sur les disques, ou l'emplacement du répertoire de chacune de ces partitions dans votre système de fichier — et encore choisir les groupes de logiciel que vous voulez installer.

© copyright Éditions Eyrolles

Installation personnalisée, ou manuelle

La suite suppose que l'on a opté pour la procédure d'installation personnalisée et propose, à chacune des étapes de base, conseils et suggestions utiles. C'est la méthode d'installation recommandée pour un serveur de base de données, de sorte que l'on peut choisir les services additionnels dont on a besoin. Il est hautement recommandé de n'inclure *aucun* autre service sur le serveur de base de données. Bien sûr, si l'on ne dispose d'aucun autre serveur pour les installer, ou bien si cela ne grève pas trop un serveur existant, on pourra quand même les ajouter pendant ce processus d'installation, de manière facile et rapide.

Créer les partitions de son choix

L'étape suivante donne un graphique des partitions du disque et propose un outil d'assignation appelé *Disk Druid*. Malheureusement, cet outil n'est disponible que pendant l'installation. Quand le système fonctionne, on peut utiliser l'utilitaire **fdisk** pour effectuer des modifications de partition, mais non l'affectation d'une partition à un répertoire.

Important. Dans un serveur de base de données, la répartition des données dans des partitions distinctes permet une meilleure gestion globale par la suite. Autant que possible, maintenir les fichiers système et les fichiers de base de données dans des partitions séparées.

Partitions d'échange multiples

L'espace d'échange est l'endroit où le système place certains ensembles de processus résidents en mémoire, quand la RAM est saturée. Il est nécessaire au bon fonctionnement du système, mais on préfère que le système ne l'utilise tout simplement *pas*, car il est d'une lenteur incroyable, comparé à la RAM. Il reste utile dans les situations d'urgence, quand la RAM est débordée, ou pour un usage occasionnel. Une utilisation continue signifie clairement que vous n'avez pas assez de RAM. Augmentez-la immédiatement si vous détectez un usage intensif de l'espace d'échange. Toute activité disque étant améliorée par la répartition des requêtes sur des disques ou contrôleurs multiples, l'espace d'échange n'échappe pas à la règle : répartissez-le pour améliorer les performances et répartissez également la charge sur tous les disques.

Supprimer une partition

Vous aurez à le faire si vous utilisez une machine qui avait déjà un système d'exploitation installé. Si vous voulez démarrer à neuf - c'est-à-dire que vous

© copyright Éditions Eyrolles

Solutions disque et mémoire

Il est techniquement possible d'utiliser des lecteurs formatés en FAT32 ou en FAT16 avec Linux, mais ce n'est pas un choix heureux pour un serveur de base de données. Le type de système de fichier optimal est le type Linux natif, valable pour tous les lecteurs de données. Il faut aussi allouer de l'espace disque pour les fichiers d'échange. L'espace d'échange peut avoir une taille maximale de 4 Go, déployée sur huit partitions, dont aucune ne peut excéder la taille de 2 Go. Il est recommandé d'allouer deux à trois fois la taille de la RAM à l'espace d'échange : une bonne règle empirique tant que l'on n'atteint pas les giga-octets de RAM. Il est peu probable que ce volume d'échanges vous concerne, voire, à un moindre niveau, une capacité de 2 Go de RAM. Il y a des exceptions : par exemple, si vous avez vraiment de gros processus insécables, configurés comme de grands fichiers de base de données résidents, il vous faut un grand espace d'échange. Si votre système est conçu pour les petites transactions orientées processus, vous pouvez avoir moins d'espace d'échange. Ne pas oublier que, si l'on a de grands processus en mémoire, le disque est économique, et l'utiliser pour les échanges. À ce point, surveiller l'activité de la RAM, en l'interrogeant à deux ou trois reprises, pour faire l'analyse du système.

n'avez pas besoin de garder ce système d'exploittation — supprimez toutes les partitions à vue. Pour conserver l'ancien système en ne faisant qu'ajouter Linux dans l'espace libre, ou réutiliser une partition libre, supprimez d'abord l'ancienne partition.

On peut supprimer toute partition existante, sur n'importe quel disque, avec *Disk Druid*. Choisir la partition dans la boîte supérieure, puis cliquer sur **Delete** et confirmer l'opération.

Modifier une partition

On peut changer la valeur d'un paramètre de la partition par la suite, à l'aide de l'option **Edit**.

Ajouter une partition

Cliquer sur le bouton **Add** pour ajouter une nouvelle partition. Il faudra le faire pour chaque partition à initialiser. S'il y a plusieurs disques physiques, choisir d'abord le disque spécifique dans la boîte inférieure.

© copyright Éditions Eyrolles

Point de montage d'une partition

Quand on clique sur **Add**, un écran de mise à jour des paramètres apparaît. Il y a quatre paramètres à saisir. Le premier paramètre est le point de montage. À la différence des systèmes fondés sur MS-DOS, qui définissent une lettre de lecteur pour chaque partition (maintenant, ce n'est plus vrai dans Microsoft Windows 2000), dans Linux et UNIX, il n'y a qu'un système de fichier, qui se sert des noms de répertoire pour faire un rapprochement et les connecter ensemble. Ces répertoires sont appelés *points de montage*. UNIX avait quelques points de montage historiques, que Linux a conservés. On peut soit entrer un nom de répertoire de son choix, soit prendre l'un des noms de répertoire préconfigurés dans la boîte à liste. La figure 2.6 montre les options de point de montage préconfigurées. Ce n'est pas une limite, mais un point de départ.

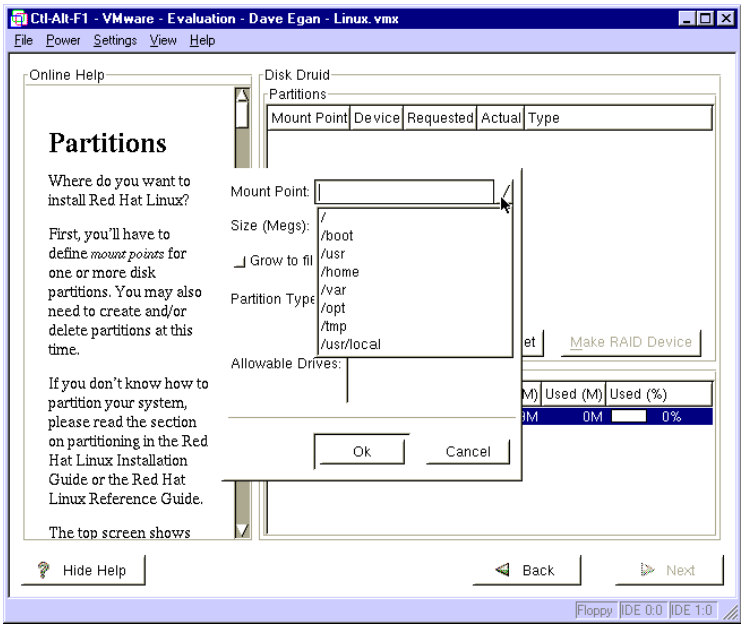
Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
Partitions
Where do you want to install Red Hat Linux?
First, you'll have to define mount points for one or more disk partitions. You may also need to create and/or delete partitions at this time.
If you don't know how to partition your system, please read the section on partitioning in the Red Hat Linux Installation Guide or the Red Hat Linux Reference Guide.
The top screen shows
Disk Druid
Partitions
Mount Point Device Requested Actual Type
Mount Point:
Size (Megs):
Grow to fil
Partition Type
/
/boot
/usr
/home
/var
/opt
/tmp
/usr/local
Make RAID Device
Allowable Drives:
Used (M) Used (%)
0M 0%
Ok
Cancel
Hide Help
Back
Next
Floppy IDE 0:0 IDE 1:0

Figure 2–6
Options de point de montage dans Disk Druid

Taille de la partition

Le second paramètre à choisir est la taille. Entrer une approximation de la taille souhaitée pour la partition, en méga-octets. A noter : la taille réelle dépend des caractéristiques du lecteur physique (la taille est arrondie au nombre entier de cylindres le plus proche) et peut s'écarter de quelques méga-octets de la taille que l'on a indiquée.

© copyright Éditions Eyrolles

L'option de taille se présente sous la forme d'un bouton « Agrandir pour remplir le disque » : cette option autorise une partition à utiliser tout l'espace disque disponible ou à partager, en quantités égales sur le même disque physique, tout l'espace disque restant, avec les autres partitions. C'est une option importante sur la dernière partition créée, ou des partitions multiples, quand il s'agit de partager tout l'espace disque restant. Elle se sert toujours de tout l'espace disque restant. On peut donner une valeur quelconque dans le champ du paramètre de taille. Disk Druid traite la taille réelle quand il effectue réellement le partage, après configuration de toutes les partitions (tout se fait en mémoire : rien ne sera réellement pris en compte tant que vous n'aurez pas validé cet écran). La figure 2.7 montre le champ de saisie de la taille, ainsi que le bouton **Agrandir pour remplir le disque**.

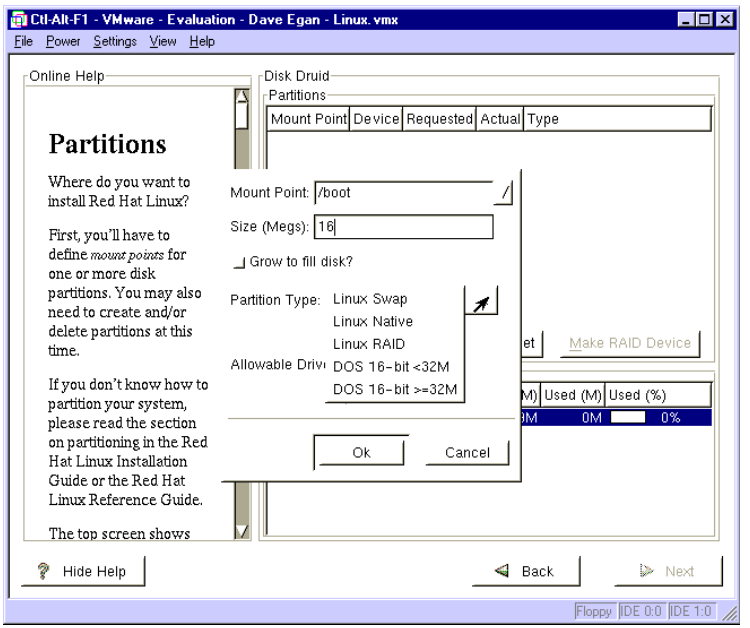
Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
Partitions
Where do you want to install Red Hat Linux?
First, you'll have to define mount points for one or more disk partitions. You may also need to create and/or delete partitions at this time.
If you don't know how to partition your system, please read the section on partitioning in the Red Hat Linux Installation Guide or the Red Hat Linux Reference Guide.
The top screen shows
Disk Druid
Partitions
Mount Point Device Requested Actual Type
Mount Point: /boot
Size (Megs): 16
Grow to fill disk?
Partition Type: Linux Swap
Linux Native
Linux RAID
Allowable Drive
DOS 16-bit <32M
DOS 16-bit >=32M
Ok
Cancel
Make RAID Device
Used (M)
Used (%)
0M
0%
Hide Help
Back
Next
Floppy
IDE 0:0
IDE 1:0

Figure 2–7
Disk Druid — Options de sélection du type de partition

Important. X Window se sert de boutons colorés à la place de la marque ✓ de Windows. Si le bouton apparaît noir (ou comme enfoncé), cela signifie **on,** ou sélectionné. La figure 2.7 illustre, pour le bouton **Agrandir pour remplir le disque**, la position **off**, de couleur claire.

Type de partition

Les partitions d'échange doivent être créées avec le type Linux Swap ; pour les autres partitions, il faut *toujours* choisir le type Linux natif. Ne choisissez

© copyright Éditions Eyrolles

pas les types DOS ; ils ne sont absolument pas optimisés pour les fichiers de base de données.

Important. Linux natif est un système de fichier très rapide, orienté fichiers de données. Linux peut utiliser DOS, HPFS, NTFS et d'autres types de partitions, mais pas de manière optimisée. Évitez-les tous, vous n'en avez pas besoin. Toutefois, de nouveaux systèmes de fichiers, à haute performance, sont en développement pour Linux et seront à prendre en considération quand ils seront prêts ; il s'agit notamment de Ext3, Reisefrs et de XFS.

Sélection d'un lecteur physique

Cette option n'est utile que si vous avez le choix entre plusieurs lecteurs physiques. Si vous n'avez qu'un seul lecteur, il est appelé **hda** (*hard disk a*). Ce sera le nom du premier lecteur physique sur le premier contrôleur de type IDE. Le lecteur suivant sera **hdb** sur le même contrôleur IDE. Sur le second contrôleur IDE, le premier lecteur sera **hdc** et le second **hdd**. Sur chacun de ces lecteurs, les partitions sont simplement numérotées de 1 à 16. À partir d'ici, la partition que le monde DOS désigne par **C:**, soit la première partition, s'appelle **hda1** dans Linux. Les partitions de type primaire suivantes seront **hda2**, puis **hda3** et **hda4**. Cependant, en raison d'une bizarrerie de la conception d'origine du schéma de partition, il en est une, dite de type primaire étendu, qui peut être **hda2**, **hda3** ou **hda4**, selon le nombre de partitions primaires qui existent déjà. Dans cette partition spéciale de type primaire étendu, on peut créer jusqu'à douze partitions additionnelles, nommées de **hda5** à **hda16**. Ainsi, techniquement, on peut avoir jusqu'à seize partitions sur un même lecteur physique de type IDE.

Pour un disque de type SCSI, le mnémonique du lecteur est **sda** ; les partitions sont nommées de **sda1** à **sda15** (Linux ne supporte que onze partitions de type « étendu », dans ce cas).

La figure 2.8 montre la table des partitions, complétées par leurs points de montage respectifs, telle qu'elle apparaîtra quand vous aurez fini de créer les diverses partitions et de leur assigner tout l'espace disque.

Technique. Il faut créer des partitions pour les besoins de gestion du système. Il y a un bon nombre de raisons, très pertinentes dans un système d'exploitation sécurisé, de découper les lecteurs de disque en partitions plus petites : facilité de gestion ; facilité de restauration restreinte à ce qui a échoué ; niveaux de sauvegarde sélectifs ; vérification sélective et gestion de quota de disque ; implémentation de politiques de sécurité.

Installation du réseau

Linux, comme UNIX, est ancré autour de la nébuleuse du protocole TCP/IP, le principal protocole de réseau, qui doit être configuré en concordance. On

© copyright Éditions Eyrolles

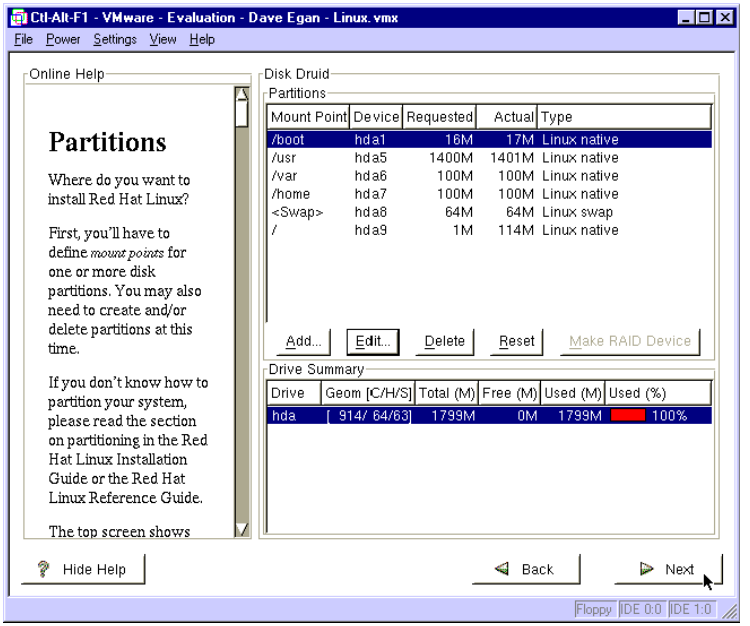
Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
Partitions
Where do you want to install Red Hat Linux?
First, you'll have to define mount points for one or more disk partitions. You may also need to create and/or delete partitions at this time.
If you don't know how to partition your system, please read the section on partitioning in the Red Hat Linux Installation Guide or the Red Hat Linux Reference Guide.
The top screen shows
Disk Druid
Partitions
Mount Point Device Requested Actual Type
/boot hda1 16M 17M Linux native
/usr hda5 1400M 1401M Linux native
/var hda6 100M 100M Linux native
/home hda7 100M 100M Linux native
<Swap> hda8 64M 64M Linux swap
/ hda9 1M 114M Linux native
Add... Edit... Delete Reset Make RAID Device
Drive Summary
Drive Geom [C/H/S] Total (M) Free (M) Used (M) Used (%)
hda [914/ 64/63] 1799M 0M 1799M 100%
Hide Help
Back
Next
Floppy IDE 0:0 IDE 1:0

Figure 2–8
Exemple de table de partition renseignée

peut sélectionner tout adaptateur réseau détecté et configurer tous les paramètres de chaque carte séparément. Sur une machine dotée d'un seul adaptateur réseau, comme celle de la figure 2.9, la première carte réseau est appelée **eth0**.

Utiliser DHCP

Le premier bouton d'option, quand on configure un adaptateur réseau, concerne l'utilisation de DHCP (*Dynamic Host Configuration Protocol*, protocole d'adressage IP dynamique) pour configurer toutes les informations relatives à TCP/IP automatiquement, à partir d'un service situé sur un serveur DHCP central : ce service fonctionne sur n'importe quel autre système d'exploitation, pourvu qu'il soit compatible avec RFC.

Activation au démarrage, ou boot

Le second bouton d'option de configuration d'un adaptateur réseau concerne la possibilité de l'activer lors du boot. Si c'est le seul adaptateur réseau, il sera sélectionné automatiquement. S'il s'agit d'une interface orientée appel entrant (*dial-in*) ou sortant (*dial-out*), on peut décider de ne pas l'activer au démarrage, et d'en faire un service activé à la demande ; mais cela sort du cadre de l'installation que nous présentons.

© copyright Éditions Eyrolles

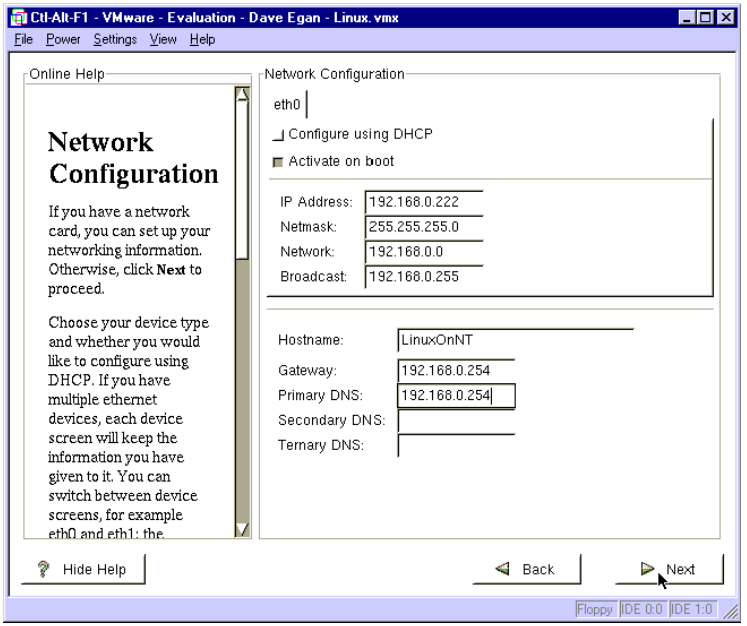
Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
Network Configuration
If you have a network card, you can set up your networking information. Otherwise, click Next to proceed.
Choose your device type and whether you would like to configure using DHCP. If you have multiple ethernet devices, each device screen will keep the information you have given to it. You can switch between device screens, for example eth0 and eth1; the
Network Configuration
eth0
Configure using DHCP
Activate on boot
IP Address: 192.168.0.222
Netmask: 255.255.255.0
Network: 192.168.0.0
Broadcast: 192.168.0.255
Hostname: LinuxOnNT
Gateway: 192.168.0.254
Primary DNS: 192.168.0.254
Secondary DNS:
Ternary DNS:
Hide Help
Back
Next
Floppy IDE 0:0 IDE 1:0

Figure 2–9
Configuration du réseau

Ne pas utiliser DHCP

Le reste de l'écran de configuration du réseau sert à configurer manuellement les options TCP/IP pour un hôte. Au minimum, il faut donner l'adresse IP et le masque de réseau associé (*netmask)*. L'interface essaie de deviner la valeur du masque de réseau si l'on utilise la touche **TAB** pour aller au champ suivant (ce calcul prend un instant très bref, ne soyez donc pas impatient). Si les valeurs estimées sont correctes, passer au champ suivant ; sinon, les corriger.

Hostname

Le *hostname*, ou nom d'hôte, n'est qu'une référence pour une machine donnée, et n'est pas nécessairement identique à la référence de cet hôte en un autre endroit du réseau. La résolution de nom est l'opération qui transforme un nom en une adresse IP. Linux se sert du fichier **/etc/hosts**, puis de DNS, pour traduire un nom d'hôte en une adresse IP avant d'émettre un message. Mais le schéma de dénomination local n'est ni partagé ni spécifique ; une autre machine peut utiliser localement le nom qu'elle désire pour l'IP de votre machine. Ce n'est que si l'on utilise un service central tel que DNS que l'on est astreint à un nom spécifique, et même, dans ce cas, cela peut être contourné. En résumé, le hostname n'a pas besoin d'être unique sur le réseau, mais il devra l'être si l'on se sert de DNS.

© copyright Éditions Eyrolles

La figure 2.9 montre la configuration d'une carte adaptateur réseau où les champs sont renseignés manuellement : adresse IP, hostname, gateway et DNS (Domain Name Service, souvent raccourci en service de nom).

Sélection du fuseau horaire

Il y a beaucoup de choses originales dans Linux, mais celle-ci est l'une des plus agréables. Tous les petits points jaunes que vous voyez représentent des noms de ville, dont certains sont très connus. Non content de cela, si l'on déplace le curseur sur la carte, une petite flèche verte apparaît, comme on le voit à la figure 2.10, qui indique le lieu-dit le plus proche de tel endroit, dans un fuseau horaire connu, tel que Gambier Islands dans le Pacifique sud. Si rien ne se passe, prenez une petite leçon de géographie avant de partir.

Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
Time Zone Selection
You can set your time zone either by selecting your computer's physical location, or by your time zone's offset from Universal Coordinated Time (also known as UTC).
Notice the two tabs at the top of the screen. The first tab offers you the ability to configure by location. With this option, you can choose your view. In choosing view, your options are: World, North America, South
Time Zone Selection
Location
UTC Offset
View:
World
System clock uses UTC
Pacific/Gambier – Gambier Islands
America/Thule
northwest Greenland
America/Thunder_Bay
Eastern Time – Thunder Bay, Ontar
America/Tijuana
Pacific Time – north Baja California
America/Tortola
America/Vancouver
Pacific Time – west British Columbi
Hide Help
Back
Next
Floppy
IDE 0:0
IDE 1:0

Figure 2–10
Configuration du fuseau horaire du système

Configuration du compte root

Toutes les distributions de Linux ont un compte spécial, appelé généralement *root*, qui est quelquefois considéré comme un *super-utilisateur*. Le compte *root* peut faire n'importe quoi sur un système Linux ou UNIX, au moins en

© copyright Éditions Eyrolles

théorie. Il faut donner son mot de passe. Sur certaines versions de Linux, on a aussi la possibilité d'installer autant d'autres utilisateurs que l'on veut. Par exemple, avec Red Hat Linux, on peut facilement ajouter un ou plusieurs utilisateurs pendant l'installation, (figure 2.11). On n'est pas obligé d'ajouter un utilisateur supplémentaire, mais il est recommandé de créer un compte sans privilège (c'est-à-dire autre que *root*), pour les travaux courants sur une station de travail. Il faut éviter d'être en *root* quand on effectue des tâches d'utilisateur de base : les erreurs commises en tant que *root* peuvent avoir des conséquences désastreuses, telles que la destruction de fichiers cruciaux, ou compromettre la sécurité et la stabilité du système

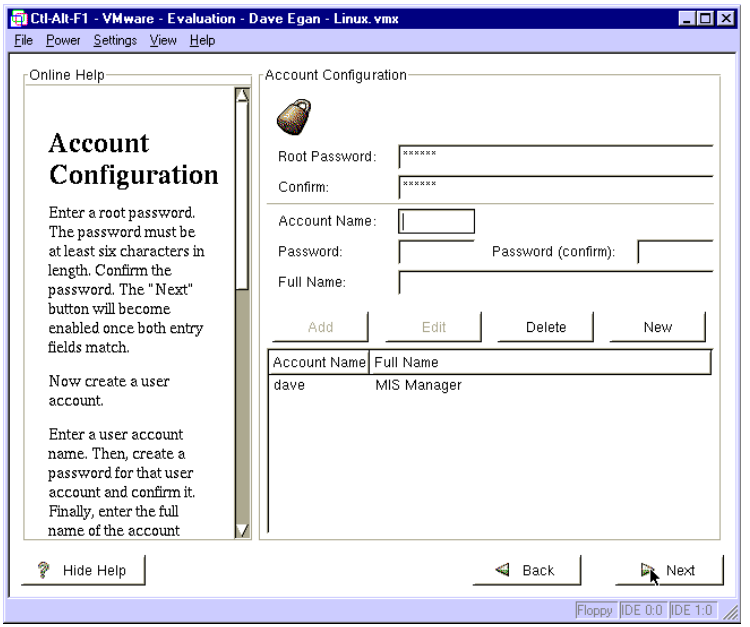

Figure 2–11
Configurer root et les comptes supplémentaires (au moins un)

Authentification

La connexion, ou login, à la plupart des systèmes Linux ou UNIX, est validée, localement ou à distance, par un service centralisé tel que NIS ou NIS+ (voir figure 2.12). Que ce soit dans l'une ou l'autre des méthodes d'authentification, on peut inclure des options complémentaires de sécurité, telles que MD5, ainsi que des mots de passe masqués, les deux étant vivement recommandés. Le MD5 est une forme de mot de passe intégrée et plus sûre que le mot de passe banal ; le mot de passe masqué est un vrai modèle de sécurité sur les

© copyright Éditions Eyrolles

fichiers de mots de passe Linux et UNIX. Normalement, Linux et UNIX conservent les informations d'installation des comptes utilisateurs dans un fichier, qui est lisible par tous les utilisateurs. Avec l'option de mots de passe masqués, certains des aspects les plus importants de cette information sont rangés dans un fichier qui n'est plus lisible par tout le monde ; il s'appelle habituellement **/etc/shadow**, mais peut avoir un tout autre nom.

Utiliser NIS ou NIS+

Le login *de domaine* centralisé standard sous UNIX et Linux est fourni par un service appelé actuellement système d'information réseau (NIS, *Network Information System*), ou une version plus courante de Sun Microsystems Inc. appelée NIS-PLUS (NIS+). Sun est le seul fournisseur de serveurs NIS+. Linux a un client compatible.

NIS propose un ensemble centralisé de fichiers de gestion des utilisateurs (similaire au PDC de Microsoft Windows NT), dupliqué sur des serveurs NIS esclaves (les BDC dans NT) qui valident le login des clients NIS (le NT LOGON de Netlogon). La figure 2.12 montre les options de NIS, sans aucune information (par défaut). Il faut connaître le nom du domaine NIS, puis soit utiliser un message pour trouver le serveur NIS esclave le plus proche, soit entrer l'adresse IP si on la connaît.

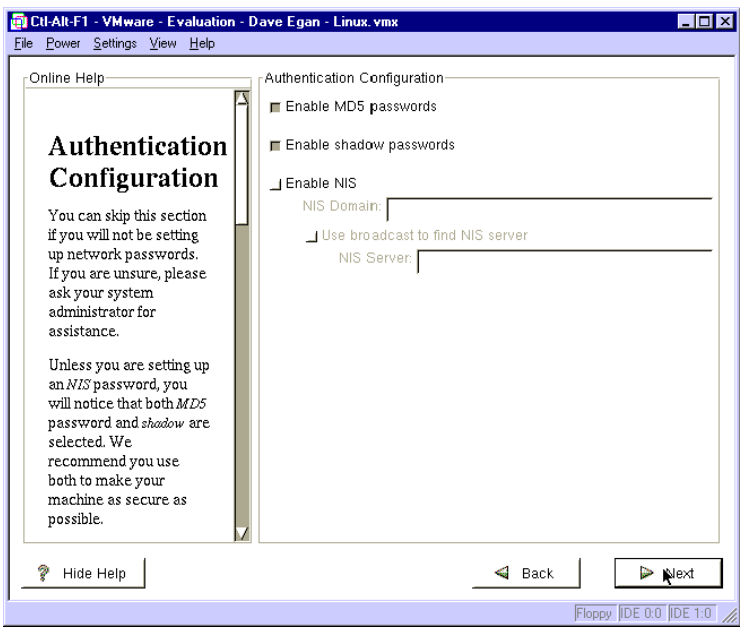
Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
Authentication Configuration
You can skip this section if you will not be setting up network passwords. If you are unsure, please ask your system administrator for assistance.
Unless you are setting up an NIS password, you will notice that both MD5 password and shadow are selected. We recommend you use both to make your machine as secure as possible.
Authentication Configuration
Enable MD5 passwords
Enable shadow passwords
Enable NIS
NIS Domain:
Use broadcast to find NIS server
NIS Server:
Hide Help
Back
Next
Floppy IDE 0:0 IDE 1:0

Figure 2–12
Configuration de l'authentification

© copyright Éditions Eyrolles

Sélection des paquetages

Il existe beaucoup de paquetages. Un paquetage représente tous les fichiers associés à un produit, ou à un composant majeur du produit. Par exemple, le paquetage du serveur web Apache contient tous les fichiers de programmes et de support de cette application. Ils sont au format RPM (*Red Hat Package Manager*), le gestionnaire de paquetage de Red Hat. Développé par Red Hat, il est disponible dans toutes les versions de Linux.

Les paquetages de base

Les paquetages de base, qui constituent le système d'exploitation Linux, sont présélectionnés et ne sont pas modifiables, dès le premier écran d'installation. On peut enlever de la sélection les paquetages de la liste qui ne sont pas nécessaires, ou ajouter tous ceux que l'on veut inclure (voir figure 2.13). Vous n'avez pas nécessairement besoin d'autres paquetages si vous envisagez d'installer le serveur de base de données ; mais ce serait une bonne idée de lire la documentation d'installation de la base de données, pour voir si certains paquetages sont nécessaires, par exemple les services de développement (C, C++, bibliothèques de développement, etc.), auxquels votre système de base de données risque de faire appel.

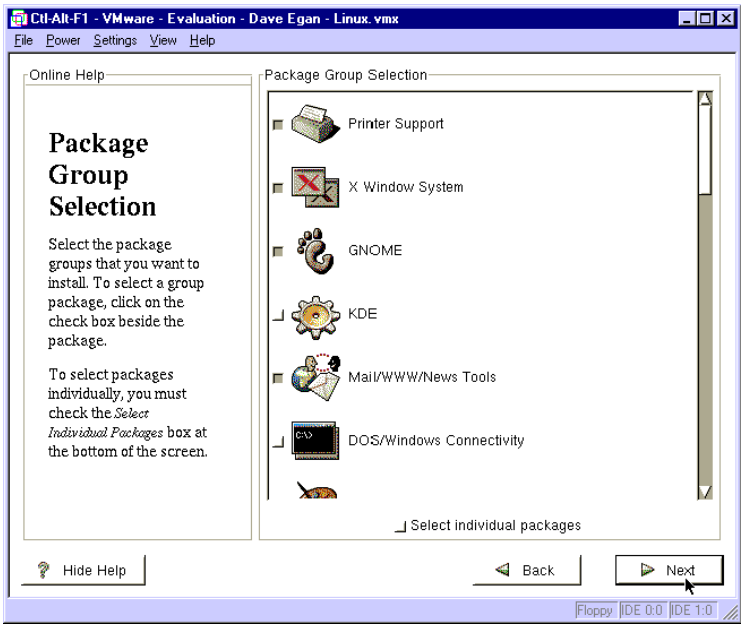
Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
Package Group Selection
Select the package groups that you want to install. To select a group package, click on the check box beside the package.
To select packages individually, you must check the Select Individual Packages box at the bottom of the screen.
Package Group Selection
Printer Support
X Window System
GNOME
KDE
Mail/WWW/News Tools
DOS/Windows Connectivity
Select individual packages
Hide Help
Back
Next
Floppy
IDE 0:0
IDE 1:0

Figure 2–13
Configuration des paquetages

© copyright Éditions Eyrolles

Services optionnels à installer

Il existe beaucoup de services optionnels : utilisateur, réseau et utilitaires. Le tableau 2.1 contient les noms des paquetages qui apparaissent dans l'écran de configuration des paquetages. Leur contenu est commenté à l'aide d'un indicateur qui précise s'il est recommandé (**Oui**), utile (**m** ou **Mn**), ou non recommandé (**n** ou **N**) pour un serveur de base de données opérationnel. Pour le cas **Mn**, des précisions sont données au paragraphe suivant.

Important. L'ordre de présentation du tableau 2.1 correspond à l'ordre que l'on voit à l'écran de la figure 2.13. Des compléments d'information sur le regroupement de ces paquetages, des suggestions d'installation et des détails sur les ensembles de paquetages sont fournis dans le tableau 2.1.

Tableau 2–1
Paquetages de Red Hat dont l'installation est optionnelle

Applications Client/Utilisateur		**Description**
Printer Support	m	Support des pilotes d'imprimante locale
X Window System	m	Système de base, nécessite Gnome ou KDE
Gnome	n	Bureau X, nécessite le système de base
KDS	n	Bureau X, nécessite le système de base
Mail/WWW/News Tools	n	Navigateurs
DOS/Windows Connectivity	n	Émulateurs et outils
Graphics Manipulation	n	Capture d'écran, manipulation, etc.
Games	N	De nombreux jeux
Multimedia support	N	Son, vidéo, etc.
Networked Workstation	Oui	Station de travail en réseau, outils réseau client
Dialup Workstation	M1	Station de travail de dialup Outils client
Network Server Services		**Services de serveur réseau**
News Server	N	Serveur de nouvelles NNTPD
NFS Server	N	Partage de fichiers UNIX et d'impression
SMB (Samba) Server	N	Partage de fichiers Win/NT et d'impression
IPX/Netware Connectivity	N	Services de fichiers Netware et d'impression
FTP Anonymous Server	N	Protocole de transfert de fichier (pour le web)
Web Server	M2	Serveur web Apache

© copyright Éditions Eyrolles

Tableau 2–1
Paquetages de Red Hat dont l'installation est optionnelle

Applications Client/Utilisateur		Description
DNS Name Server	N	BIND 8 (pour Internet/intranet)
PostgreSQL Server	M3	Serveur de base de données ** (gratuit !)
Development Packages		**Paquetages de développement**
Network Management Station	Oui	Station de gestion de réseau : outils et SNMP
TeX Document Formatting	N	Formatage de document TeX : traitement de texte, publication
Emacs	N	Le meilleur environnement d'édition
Development	M4	Compilateurs et bibliothèques
Kernel Development	M5	Pour reconstruire le noyau
Extra Documentation	Oui	HOWTO, documents complémentaires, aide en ligne
Utilities	Oui	D'autres fonctions du système

Les fonctionnalités optionnelles

Voici quelques-uns des programmes et des caractéristiques optionnelles disponibles à prendre en considération lors de l'installation :

- **M1.** Dialup en réseau peut être utile pour le diagnostic et la maintenance des appels entrants à distance.
- **M2**. Avec le serveur web, vous pouvez gérer votre système à l'aide d'une interface web, mais cela est de votre ressort — ou de celui de l'application que vous installez.
- **M3**. Si vous ne voulez pas acheter une application de serveur de base de données compatible SQL, utilisez Postgres, le seul système gratuit.
- **M4**. Vous serez peut-être amené à recompiler certains aspects de votre nouvelle application de serveur de base de données et aurez besoin des applications de développement ; vérifiez votre documentation.
- **M5**. Il se peut que le noyau par défaut de la plupart des installations Linux que vous installez pour la première fois ne soit pas optimisé pour votre matériel.

Remarque. Pour vraiment optimiser la plupart des versions de Linux, il faut exécuter les utilitaires de reconfiguration du noyau, qui améliore le support des processeurs de

© copyright Éditions Eyrolles

niveau avancé (livré par défaut au niveau 386, il passe au niveau 586, PII, PIII, Athlon, etc.) et le support des SMP (matériel multi-processeur symétrique), RAID, SCSI, etc. Il vous faudra tout lire sur ce sujet. Est-ce absolument nécessaire ? *Non, mais c'est fortement conseillé*, car, une fois bien expliqué, ce n'est pas difficile. La documentation complémentaire donne toutes les bases nécessaires pour la reconstruction (recompilation) du noyau et son installation. Tout livre assez épais sur votre distribution de Linux, du genre « Tout ce que vous voulez savoir à propos de » couvrira ce sujet dans des détails fastidieux, mais ce sera un bon investissement.

Option « Tout installer »

Il est encore possible de tout installer. Si vous avez de l'espace disque en abondance (environ 1,5 Go sous la partition **/usr),** prenez cette option plutôt que de décider au cas par cas ce qui est nécessaire. Un mot au passage, cependant ; certains des paquetages installés tenteront de lancer des démons au démarrage du système et installeront des fichiers susceptibles de poser des problèmes de sécurité. Les utilisateurs, et même les administrateurs, doivent se familiariser eux-mêmes avec les paquetages et contrôler tout ce qu'ils font. Par exemple, autoriser le montage NFS sur un serveur exposé à l'Internet ouvre une brèche significative dans la sécurité du système. Il y a une certaine sagesse à n'installer que ce dont on a besoin sur un système critique.

Installation de X Window (optionnelle)

C'est probablement la partie de l'installation la plus difficile à comprendre pour un utilisateur moyen de PC. Le système X Window est en réalité un programme de niveau utilisateur, et non de niveau noyau, et peut accéder directement au matériel de la carte vidéo. Dans un environnement Windows, il y a un pilote pour le contrôleur vidéo. De même, on trouve un fichier de configuration du pilote, en mode texte dans ce cas, qui pointe vers le pilote vidéo et initialise quelques options de configuration, notamment les options de définition.

Vous n'avez pas besoin d'avoir X Window pour exécuter Linux, de plus il consomme beaucoup de ressources CPU. Sur un serveur réel dédié, vous n'installerez pas X Window. Comme à la figure 2.14, choisissez l'option **Skip X configuration**, sauf si vous avez inclus le paquetage de X Window et au moins un bureau, Gnome ou KDE.

La suite donne un aperçu rapide de la manière d'installer X Window sur un système de base. Le système sera probablement capable de détecter la carte vidéo, sauf s'il s'agit d'un matériel de technologie vraiment récente, mais, en général, la plupart des puces vidéo connues seront découvertes automatiquement, comme dans Windows.

© copyright Éditions Eyrolles

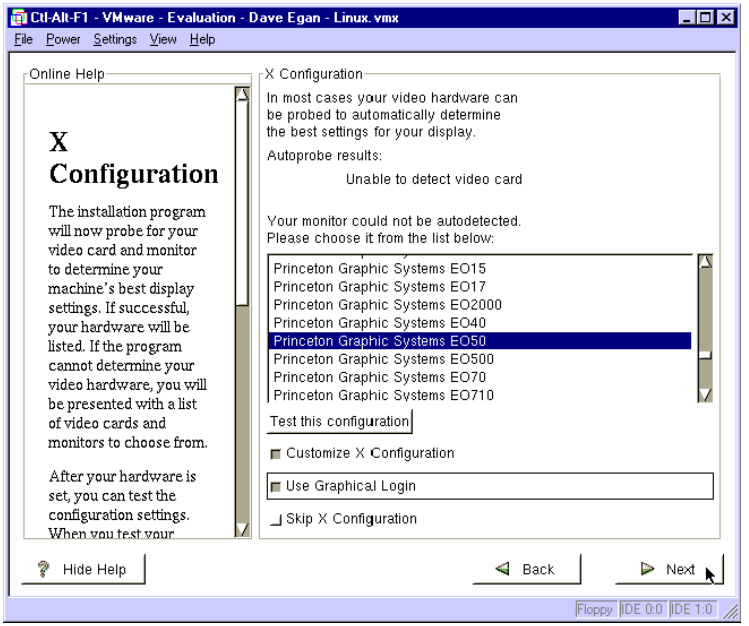
Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
X Configuration
The installation program will now probe for your video card and monitor to determine your machine's best display settings. If successful, your hardware will be listed. If the program cannot determine your video hardware, you will be presented with a list of video cards and monitors to choose from.
After your hardware is set, you can test the configuration settings. When you test your
X Configuration
In most cases your video hardware can be probed to automatically determine the best settings for your display.
Autoprobe results:
Unable to detect video card
Your monitor could not be autodetected. Please choose it from the list below:
Princeton Graphic Systems EO15
Princeton Graphic Systems EO17
Princeton Graphic Systems EO2000
Princeton Graphic Systems EO40
Princeton Graphic Systems EO50
Princeton Graphic Systems EO500
Princeton Graphic Systems EO70
Princeton Graphic Systems EO710
Test this configuration
Customize X Configuration
Use Graphical Login
Skip X Configuration
Hide Help
Back
Next
Floppy IDE 0:0 IDE 1:0

Figure 2–14
Optionnel — configuration de X Window, partie 1

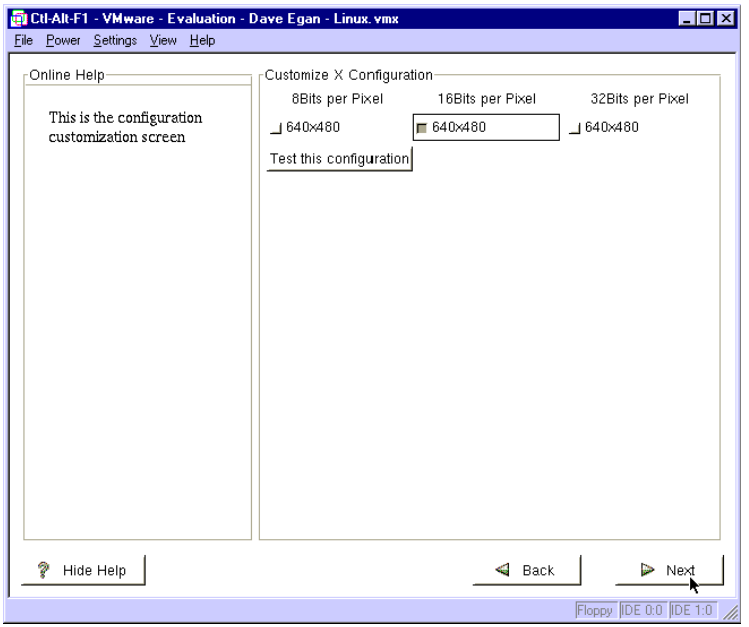

Figure 2–15
Optionnel — configuration de X Window, partie 2

© copyright Éditions Eyrolles

Logiciel serveur X

Les paquetages de serveur X Window varient selon le type de matériel : VGA générique ou SVGA, et selon les puces vidéo bien connues de type S3 ou ATI. Si la recherche aboutit, les noms de la carte vidéo et de la puce seront affichés. Sinon, vous pouvez choisir le matériel dans la liste fournie. Les cartes vidéo génériques reposent normalement sur l'une des puces vidéo standard telles que S3, ATI, etc. Si cela ne marche pas, en particulier sur les portables, choisissez le pilote SVGA générique et le moniteur Multi-sync générique. Cela doit fonctionner sur la plupart des matériels de base, mais il n'y a aucune garantie. Il existe un site dédié entièrement aux fichiers de configuration des divers portables, qui ont toujours au moins quelques matériels propriétaires, mal identifiés. Pour plus d'informations, consultez les sites web suivants :

- www.XFree86.org pour des informations à propos du serveur X ;
- www.cs.utexas.edu/users/kharker/linux-laptop/ pour le support des portables.

Modes vidéo X

Après avoir sélectionné le pilote vidéo X correct, on choisit le mode vidéo, comme dans Windows. On aura le choix entre diverses profondeurs de couleur — 8, 16, ou 32 bits par pixel — et entre diverses définitions allant de 640 x 480 jusqu'au niveau le plus élevé supporté par la carte (figure 2.15). Par défaut, c'est la résolution la plus élevée qui peut être supportée sans aucune autre option. On peut la modifier de manière à basculer entre les configurations par la combinaison des trois touches Ctrl-Alt-+. Pour plus de détails, consultez le guide d'installation ou la documentation de Linux.

Installation réelle physique

À ce stade, vous avez fourni toutes les informations nécessaires à l'installation et vous êtes maintenant prêt à démarrer le formatage réel du système de fichiers et l'installation des paquetages sélectionnés. L'écran suivant affiche le logo Red Hat et propose soit de revenir en arrière et de faire des modifications (ce n'est pas recommandé : il vaut mieux redémarrer et recommencer), soit de cliquer sur **Next** pour continuer.

La figure 2.16 montre le formatage du système de fichier (dans le format e2fs Linux, et non en FAT DOS ou NTFS) tel qu'il apparaît sur l'écran d'installation, avant que l'installation réelle ne démarre. La figure 2.17 est un écran final qui attend que l'on clique sur l'option **Next**, avant de commencer l'installation réelle des fichiers.

© copyright Éditions Eyrolles

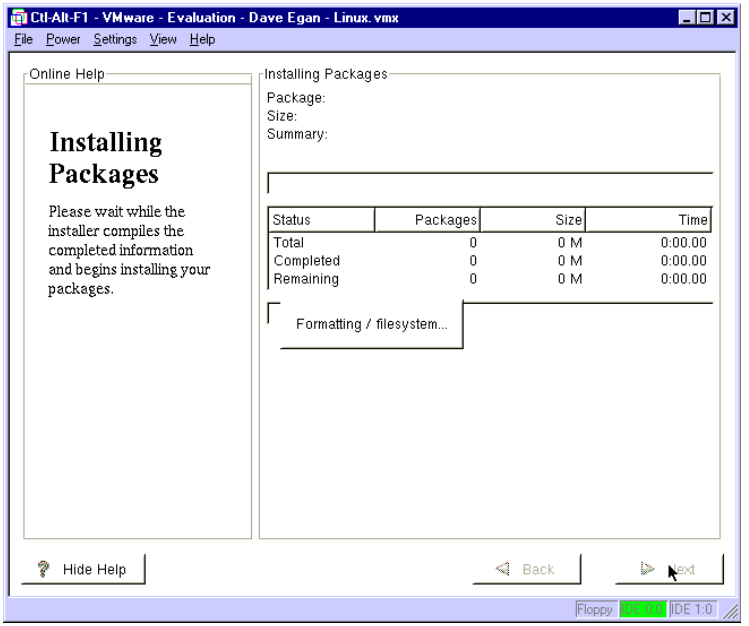

Figure 2–16
Formatage des partitions du système de fichiers avant l'installation

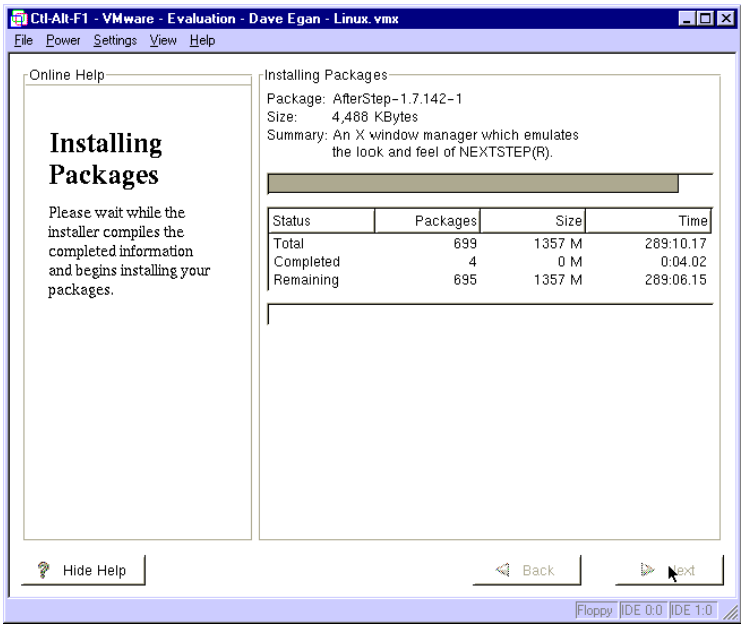
Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
Installing Packages
Please wait while the installer compiles the completed information and begins installing your packages.
Installing Packages
Package: AfterStep-1.7.142-1
Size: 4,488 KBytes
Summary: An X window manager which emulates the look and feel of NEXTSTEP(R).
Status Packages Size Time
Total 699 1357 M 289:10.17
Completed 4 0 M 0:04.02
Remaining 695 1357 M 289:06.15
Hide Help
Back
Next
Floppy IDE 0:0 IDE 1:0

Figure 2–17
Écran d'installation des paquetages

© copyright Éditions Eyrolles

Installation des paquetages

La figure 2.17 montre une installation à 90 % au cinquième paquetage. Remarquez que le temps d'installation estimé est supérieur à 289 heures ! Cependant, dès que les premiers paquetages sont installés, le programme affiche un temps d'installation de plus en plus précis. Une installation dure en principe de cinq à vingt minutes sur un matériel de type courant, et dépend de la vitesse d'entrée/sortie du système disque, ainsi que du nombre de paquetages à installer.

Installer Debian en jouant à Tetris

Si vous êtes en train d'installer une distribution Debian, vous pourrez jouer à Tetris tandis que les paquetages s'installent. Une délicate attention, spécialement sur les équipements les plus anciens, sur lesquels l'installation des paquetages dure un certain temps.

Linux Loader (LILO) et disquette de démarrage

Il est conseillé de se servir de Linux Loader et d'une disquette de boot — une seule suffira. On dispose ainsi d'une amorce alternative aux fichiers système, qui passe outre la partition active courante. On peut créer cette disquette de boot à tout moment à l'aide de l'utilitaire **mkbootdisk** de Linux Red Hat (le même ou son équivalent dans d'autres versions ; voir la documentation). La figure 2.18 montre l'écran proposé : mettez une disquette dans le lecteur (recommandé), ou bien annulez l'opération.

Configuration de LILO

Si l'on a installé Linux alors qu'un système d'exploitation graphique était déjà installé sur le système, choisir l'OS de démarrage par défaut, sachant que l'installation Linux prend la partition de boot Linux dans tous les cas.

Disque de boot d'urgence

Il est aussi vivement conseillé de créer une disquette de boot d'urgence. C'est une solution de *remplacement*, valable uniquement pour le bloc de boot, un moyen de contournement en cas de corruption de la partition de boot (si le système perd son secteur de boot pour une raison quelconque, insérez cette disquette et redémarrez le système). Cette disquette de boot sert encore à la restauration d'urgence, si elle est accompagnée d'une seconde, créée avec le fichier **RESCUE.IMG,** comme nous l'avons expliqué auparavant.

© copyright Éditions Eyrolles

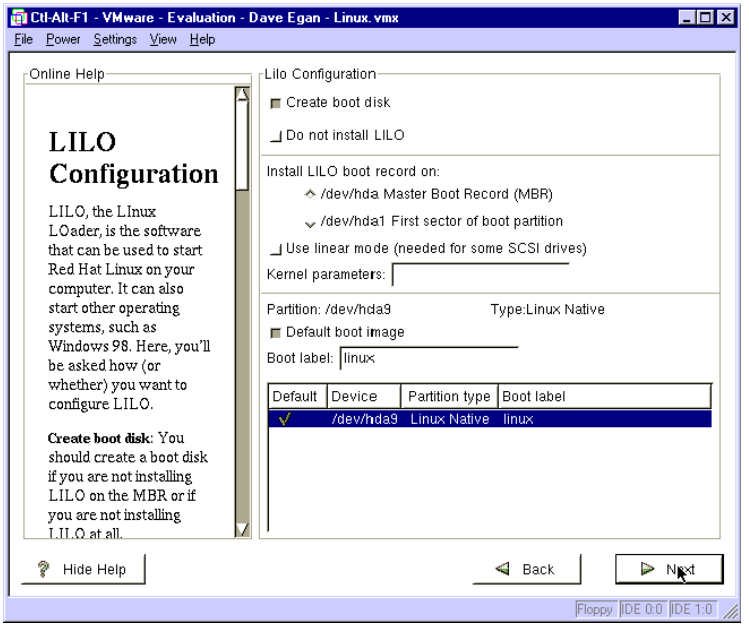
Ctl-Alt-F1 - VMware - Evaluation - Dave Egan - Linux.vmx
File Power Settings View Help
Online Help
LILO Configuration
LILO, the LInux LOader, is the software that can be used to start Red Hat Linux on your computer. It can also start other operating systems, such as Windows 98. Here, you'll be asked how (or whether) you want to configure LILO.
Create boot disk: You should create a boot disk if you are not installing LILO on the MBR or if you are not installing LILO at all.
Lilo Configuration
Create boot disk
Do not install LILO
Install LILO boot record on:
/dev/hda Master Boot Record (MBR)
/dev/hda1 First sector of boot partition
Use linear mode (needed for some SCSI drives)
Kernel parameters:
Partition: /dev/hda9
Type:Linux Native
Default boot image
Boot label: linux
Default Device Partition type Boot label
/dev/hda9 Linux Native linux
Hide Help
Back
Next
Floppy IDE 0:0 IDE 1:0

Figure 2–18
Configuration de LILO et création d'une disquette de boot (à tenter !)

Installation terminée !

Vous avez pratiquement terminé ! Redémarrage ! Le dernier écran vous félicite pour votre installation et vous demande d'appuyer sur la touche **ENTER** pour redémarrer. Ne pas éteindre à ce moment, car il faut arrêter le système proprement. En fait, il faut effectuer un arrêt normal à chaque fois que l'on veut éteindre la machine, sinon on risque des problèmes de corruption de fichiers. La plupart des bases de données ont aussi leur propre procédure d'arrêt normal qui vous épargnera bien du temps et des risques.

Configurer le serveur pour une base de données

Trouver un bon administrateur et lui demander de le faire. Sinon continuer l'installation et organiser l'espace disque selon les besoins : volumes, répartition, utilisation du RAID, sauvegardes fréquentes, etc. La suite de cet ouvrage donne plus de détails sur la manière d'initialiser Linux en tant que serveur de base de données. Pour l'essentiel, il faudra configurer les interfaces réseau, créer ou restaurer tous les fichiers de données nécessaires, tester l'accès de tous les clients à travers le réseau et initier le schéma de sauvegarde.

© copyright Éditions Eyrolles

Points à vérifier

Les disques, la RAM, le processeur et les cartes réseau sont les caractéristiques les plus déterminantes pour les performances. Tentez l'aventure sans utiliser cependant de nouveau matériel : partout où c'est possible, n'utiliser que du matériel vraiment testé.

Il faut éviter deux écueils en matière de partitions : en créer trop peu, ou trop. Ne créer initialement que ce dont on a besoin, quitte à augmenter le nombre de partitions si besoin est. La pire des erreurs est de ne pas allouer assez d'espace aux partitions critiques du système de fichiers, telles les partitions **root**, **/** et la partition **/var (**s'il doit y avoir beaucoup de connexions et de vérifications). L'application de base de données devra se trouver sur une partition séparée de l'OS et des données. Les données seront aussi sur leur propre partition — ou sur des partitions multiples distinctes — selon la conception du système. En outre, si chaque zone critique se trouve dans une partition séparée, on peut facilement la transférer plus tard dans une nouvelle partition, plus grande, et adapter plus simplement le système à l'augmentation des besoins. Les partitions distinctes sont plus faciles à sauvegarder et à restaurer, et beaucoup plus faciles à gérer en termes de quotas de nombre d'utilisateurs et de sécurité.

Autres versions de Linux

La section précédente s'est concentrée sur une installation de Red Hat, mais on peut évidemment utiliser l'une ou l'autre des distributions Linux, qui seront similaires en termes de performance, de caractéristiques et de stabilité. La seule différence se situe dans l'installation initiale. Il peut y avoir ensuite de grandes différences dans les utilitaires de gestion du système, qui resteront néanmoins cantonnées à leur présentation. La gestion de Linux ne change en aucune manière au niveau de base. On peut toujours ajouter des utilisateurs, des configurations de réseau, des configurations de système, et des services de la même manière à ce niveau, bien que les interfaces soient différentes.

Les autres versions de Linux ont été exposées au chapitre 1. Une exception notable, la version Stampede Linux, que l'on installe *via* une connexion réseau et qui est optimisée pour chaque matériel. L'installation exige une connexion réseau sur Internet et un type **sysadmin** UNIX qui peut déchiffrer les requêtes d'installation typiques d'UNIX. Ce style d'installation n'est pas d'un maniement facile ; elle télécharge le code source puis le compile avec les options spécifiques du matériel, de sorte que le système fonctionne de manière optimale. Pour plus d'informations, voir www.stampede.org.

© copyright Éditions Eyrolles

Exemples de scénarios de base de données

Voici quelques exemples de scénarios réels qui implémentent des bases de données sur Linux, avec les ressources matérielles conseillées.

Scénario 1

Q. Vous voulez installer un serveur de base de données Oracle, par exemple : il vous faut d'abord installer un serveur Linux. La base de données fera 50 Mo, avec une croissance prévue de 50 Mo par an. Vous devez aussi fournir des services web, à partir de ce serveur, à 100 utilisateurs. De quoi avez-vous besoin ?

R. Vous devez avoir au moins 256 Mo de RAM sur ce serveur et, si possible, un RAID matériel (à défaut, un RAID logiciel) sur environ 2 Go d'espace disque, pour les transactions et les connexions. Il faut une carte réseau de 10 à 100 mégabits. Si une seule machine dotée d'un processeur de 500 MHz ou plus suffit pour la première année, elle sera inadaptée par la suite. Prévoir une seconde machine, si possible.

Scénario 2

Q. Vous installez Informix : vous avez une base de données de 800 Mo qui atteindra 3 Go cette année, et qui doublera tous les six mois par la suite, avec 500 utilisateurs qui font la plupart du temps de brèves requêtes d'interrogation, et environ 100 utilisateurs qui ajoutent des enregistrements en permanence. De quoi avez-vous besoin ?

R. Il vous faut au moins 4 Go de RAM, 4 Go d'espace d'échange, et au moins deux processeurs équipés d'une grande *ferme* de disques — au moins 10–12 giga-octets — pour gérer en direct des sauvegardes locales, beaucoup de connexions et de transactions, et un espace de journalisation. Vous pouvez choisir un serveur équipé de processeurs Alpha pour obtenir la plus grande taille de mémoire virtuelle : cela permet de mettre la base de données entière en mémoire sans espace d'échange. Si vous dépassez une taille de 2 Go pour le processus utilisateur, il vous faudra un serveur à base d'Alpha à coup sûr.

© copyright Éditions Eyrolles

Résumé

Ce chapitre vous a familiarisé, étape par étape, avec une installation rapide en mode graphique d'un système Red Hat. Nous avons retenu Red Hat parce que c'est la distribution Linux dominante sur le marché.

- La première chose à faire est de bien concevoir le système et d'évaluer les besoins, de manière à définir le matériel correspondant.
- Il faut ensuite s'assurer que l'on dispose du matériel adéquat, et surtout orienté serveur de base de données — de préférence avec des processeurs multiples, beaucoup de RAM, et un sous-système d'entrée/sortie disque sous-jacent rapide, doté de beaucoup d'espace disque.
- Il est bon d'avoir à l'esprit les concepts des cartes mères des PC Intel sur lesquels se fondent les partitions, la limite de 16 partitions sur un lecteur de type IDE, et de 15 partitions sur un lecteur SCSI.
- Vous devez être soigneux au moment de la partition du système de fichiers : la plupart des versions de Linux et d'UNIX attendent des partitions relativement petites, faciles à gérer.
- Les partitions critiques pour le système sont **/**, **/usr**, **/var**, **/home**, **/usr**, **/opt**, et **swap**, pour n'en citer que quelques-unes.
- Les options d'installation vous permettront de conserver le système d'exploitation courant et de disposer d'un double boot ; certaines options peuvent effacer les partitions des autres systèmes. Les serveurs de production doivent en principe être dédiés à Linux, et ne pas utiliser de partitions non Linux.
- Les paquetages à installer sont nombreux : n'installer que les paquetages nécessaires.
- Créez la disquette de boot à la fin de l'installation.
- N'utilisez pas le serveur de base de données en tant que station de travail utilisateur ; il sera dédié uniquement aux services de base de données, en vue d'une meilleure performance.
- Les outils de gestion de système varient considérablement dans Linux et UNIX en général ; garder les livres fournis à portée de la main.
- Il est vivement conseillé de faire appel à quelqu'un qui a une expérience de l'administration de système UNIX ou Linux, car les compétences d'administration des systèmes Windows ne sont pas transposables.

Si vous n'avez d'expérience ni sur Linux ni sur UNIX, vous avez un monde nouveau tout entier à explorer du point de vue du matériel, du logiciel et des interfaces. Plongez-y et jouez, mais utilisez une machine de test pour cela.

© copyright Éditions Eyrolles

FAQ

Q. J'ai un disque plutôt ancien, puis-je l'utiliser ?

R. Non, c'est à éviter. Dans beaucoup de cas, les disques anciens n'utilisent pas les caractéristiques IO avancées de la carte mère et vont ralentir considérablement votre machine.

Q. On m'a dit que l'on peut exécuter Linux à partir de Windows. Est-ce vrai ?

R. Oui, il y a un produit très valable appelé VMWare (www.vmware.com), qui permet d'utiliser presque tous les systèmes d'exploitation (tels Linux ou NT) en tant que système d'exploitation hôte, dans lequel on peut installer un ou plusieurs systèmes d'exploitation virtuels additionnels (tels Win9x, NT, et Linux). Les copies d'écran utilisées dans ce manuel ont été obtenues lors d'une installation de Linux sur une machine d'hôte NT 4.0 dont tous les lecteurs étaient déjà formatés en FAT et NTFS. VMWare a créé un disque virtuel sur l'un des volumes FAT qui disposait tout juste de 1,8 Go d'espace libre. Une installation complète de Red Hat Linux dans laquelle six partitions simulées ont été créées, comme on l'a vu dans les figures montrées auparavant, a utilisé 1,3 Go de la partition FAT (probablement en format compressé) pour le disque virtuel. N.B. : le matériel recommandé est au moins un PC à 266 MHz avec 96 Mo RAM et, comme dans toute recommandation, il s'agit d'un minimum. Ayez beaucoup plus de RAM et de puissance si vous pouvez, ou préparez-vous à une lente traversée des vicissitudes de l'installation (16 heures sur un 166 MHz avec 64 Mo RAM).

Q. Pourquoi choisir Red Hat ?

R. Red Hat est un bon choix parce que c'est le système le plus connu, qu'il fournit des services d'assistance et de formation, et qu'il est en tête des produits shareware.

Q. L'assistance est-elle un poste important ?

R. Cela dépend. On peut envisager de payer des services d'assistance, sous forme de prestations, ou préférer améliorer les compétences d'un service interne, pour le même coût à peu près.

Q. Que faire si je n'ai aucune connaissance d'UNIX ?

R. L'Internet est, à plus de 90 %, fondé sur UNIX, et ce n'est pas un désastre d'en faire une première expérience. Le film *Toy Story* a été entièrement généré par ordinateur sur des machines UNIX. Le film *Titanic* a utilisé Linux sur des serveurs équipés de processeurs Alpha, pour réaliser tous les effets spéciaux. UNIX est partout — mais il ne fait pas la course au marché des PC, comme le font les concurrents.

© copyright Éditions Eyrolles

Q. J'ai entendu dire que l'installation est vraiment épouvantable ! Est-ce vrai ?

R. Si vous êtes un utilisateur moyen de PC qui prend une machine toute configurée pour fonctionner, oui, vous aurez un peu la tête dans le sac. Mais avec le foisonnement actuel d'installations graphiques pour Linux, vous devriez vous en tirer et effectuer une installation de base sans trop de difficulté.

© copyright Éditions Eyrolles

3

Installer et utiliser Oracle sur Linux

Dans ce chapitre

- Concepts
- Installer Oracle8
- Installer Oracle8i
- Utiliser Oracle8/8i

© copyright Éditions Eyrolles

Introduction

Oracle a la plus grande part du marché mondial des logiciels de gestion de base de données. Oracle a atteint cette position dominante en produisant et en faisant évoluer l'un des meilleurs systèmes de gestion de base de données relationnelle (SGBDR) existants : puissant, souple et modulable à l'extrême. Le SGBDR Oracle n'est pas un logiciel en développement libre (*open-source*) ; par contre, c'est une application très stable et solide une fois qu'elle est correctement installée et configurée. Que ce soit avec Linux ou d'autres plates-formes puissantes, Oracle est hautement configurable et modulable, ce qui peut sembler compliqué à qui n'en est pas familier. Contrairement à d'autres produits de serveur de base de données SQL, Oracle n'est pas un logiciel à « taille unique ».

Oracle8 présente une richesse fonctionnelle incroyable et il faudrait des pages et des pages d'explications pour exposer toutes les caractéristique de cette base de données. Le but de ce chapitre est donc de vous aider à installer et à exécuter Oracle8 ou Oracle8i sur Linux, vous pourrez ensuite chercher ailleurs une documentation plus avancée. Ce chapitre présente un aperçu de la manière d'installer, de configurer et d'utiliser la base de données Oracle sur un serveur Linux. Plusieurs solutions propres à Linux sont abordées ; mais, pour un traitement plus fin de la base de données Oracle, il faut faire appel aux nombreuses publications sur ce logiciel. Au moment où est publié cet ouvrage, il existe deux versions du SGBDR Oracle, Oracle8 et Oracle8i. Chacune de ces versions est disponible en deux éditions différentes : Standard et Entreprise. L'édition Entreprise d'Oracle8 offre plus de fonctionnalités et d'options que l'édition Standard.

Oracle8 est aussi connu comme la version 8.0.5, et Oracle8i comme la version 8.1.5. L'une des caractéristiques les plus importantes qui différencie Oracle8 et Oracle8i est que Oracle8i comprend un environnement d'exécution Java optionnel intégré dans le serveur. Cette caractéristique permet d'écrire des procédures en Java, exécutées par le serveur de base de données. Oracle8i comprend aussi de nouvelles applications Java dont un programme d'installation, un assistant de configuration de base de données et des utilitaires de configuration réseau. Ces programmes facilitent généralement son installation et son utilisation. Les applications Java sont nouvelles et ne s'exécutent pas toujours comme on s'y attendrait sous Linux. Certaines personnes rencontrent quelques difficultés à installer correctement l'environnement d'exécution Java, ou il arrive qu'une condition inattendue interrompe l'application. La version 8.1.6 d'Oracle 8i pour Linux, appelée Oracle8i version 2, sort, en principe, au premier trimestre 2000 : elle comprend des caractéristiques supplémentaires et la correction des bogues.

© copyright Éditions Eyrolles

On sait que l'installation d'Oracle n'est pas une petite affaire, mais une fois Oracle installé, vous aurez l'une des meilleures bases de données fonctionnant sur Linux. Les instructions d'installation présentées dans ce chapitre concernent Oracle8 et Oracle8i, qui ont beaucoup d'étapes communes d'installation. Lisez-les attentivement : elles sont quelque peu complexes, du fait qu'elles couvrent les deux versions. Il y a quelques différences importantes entre les installations, mais, en général, elles sont similaires. Les instructions d'installation concernent l'édition standard pour Oracle8 et l'édition Entreprise pour Oracle8i. L'édition standard est moins chère que l'édition Entreprise, et comprend moins de caractéristiques et d'options. Laquelle installer ? Visitez le site Internet d'Oracle (www.oracle.com) pour comparer les deux produits. Si vous êtes intéressé par un moteur Java, inclus dans le serveur de base de données, par l'utilisation de XML ou par les utilitaires conviviaux écrits en Java, Oracle8i est le bon choix. Pour un environnement simple, qui s'appuie sur les dernières ressources, Oracle8 conviendra à votre besoin. L'édition standard présente suffisamment de caractéristiques pour occuper un débutant.

Les sections suivantes présentent une description des concepts les plus importants et des termes nécessaires pour se familiariser avec le logiciel Oracle, les instructions d'installation d'Oracle8 et Oracle8i, une vue d'ensemble des outils de base du maniement de la base de données et, enfin, une brève introduction à quelques applications tierces.

Concepts

Avant de se plonger dans les détails de l'installation, il faut comprendre certains concepts. Les systèmes de bases de données sont souvent répartis en trois catégories : les systèmes transactionnels (OLTP), les systèmes d'aide à la décision (DSS), et les systèmes hybrides. OLTP se caractérise par de courtes requêtes ou mises à jour de la base de données, par exemple le système de réservation d'une compagnie aérienne. Les systèmes d'aide à la décision sont caractérisés par des requêtes longues à s'exécuter, comme le rassemblement des informations météorologiques. Un système hybride comprendra un mélange des deux ; par exemple, une application commerciale de type web qui peut prendre des commandes (OLTP) et fournir des services de recherche d'informations (DSS). Ces trois classifications représentent des modèles de base qui orientent la conception et la configuration de la base de données.

Il existe une distinction importante entre une *base de données* Oracle et une *instance* Oracle. La base de données, identifiée par le nom de la base de données (DB_NAME), est composée de fichiers physiques utilisés pour le

© copyright Éditions Eyrolles

stockage. Une instance, identifiée par l'identificateur système (SID), est constituée des structures de mémoire et des processus utilisés pour répondre pleinement aux requêtes des utilisateurs. Quand un serveur Oracle s'arrête, l'instance cesse d'exister. La raison fondamentale pour laquelle on les différencie est qu'une même base de données peut être servie par des instances multiples. La base de données et l'instance sont constituées de nombreux composants. Les structures de mémoire comprennent la zone globale système, le pool partagé, la mémoire cache de la base de données, le cache de journalisation et la zone globale des programmes. Les structures logiques comprennent les tablespaces, les schémas, les segments, les extents, les segments de rollback, les tables et les index. Les composants physiques comprennent les fichiers de données, les fichiers de contrôle, les fichiers journaux et les fichiers de paramètres.

Zone globale système

Les structures de mémoire d'une instance Oracle sont rangées dans une région de la mémoire appelée la zone globale système (SGA, *System Global Area*). La SGA est allouée dans la mémoire virtuelle, où réside le processus du serveur Oracle. Elle comprend plusieurs structures de mémoire dont le pool partagé, la mémoire cache de la base de données et le cache de journalisation. De nombreux processus partagent la SGA.

Pool partagé

Le pool partagé contient deux composants : le cache de bibliothèque et le cache du dictionnaire des données. Le cache de bibliothèque stocke les instructions SQL le plus récemment utilisées et leurs plans d'exécution. Le cache du dictionnaire des données stocke les informations du dictionnaire des données le plus récemment utilisées telles que les définitions de table, les noms d'utilisateurs et les privilèges. La taille du pool partagé peut affecter les performances de la base de données, notamment dans un environnement OLTP.

Cache mémoire de la base de données

Quand une requête SQL est soumise par un utilisateur, le processus serveur cherche dans la mémoire cache de la base de données les blocs de données nécessaires pour répondre à la demande. S'ils ne sont pas trouvés dans le cache, le processus serveur doit lire les blocs à partir d'un périphérique physique et placer une copie des données dans la mémoire cache. Les requêtes suivantes qui porteront sur les même blocs seront traitées de la même manière et ne nécessiteront pas de nouvelles lectures physiques.

© copyright Éditions Eyrolles

Cache de journalisation

Tous les changements effectués sur les données sont stockés dans le cache de journalisation, et seront copiés plus tard dans des fichiers journaux physiques.

Zone globale des programmes

La zone globale des programmes (PGA, *Program Global Area*) est une région de mémoire qui contient les données d'un seul processus serveur. Quand on l'utilise dans une configuration de serveur dédiée, la PGA contient une zone de tri, des informations sur les sessions, l'état des curseurs et une zone de pile. La PGA est allouée à un processus quand il démarre et désallouée quand le processus est terminé.

Tablespaces

Une base de données est divisée en unités de mémoire logique appelées tablespaces. Un *tablespace* regroupe des structures logiques liées entre elles et se présente sous forme d'un ou plusieurs fichiers physiques. Il est d'une pratique courante de grouper certains types d'objets dans un tablespace : par exemple, les tables temporaires, les segments de rollback ou les tables du dictionnaire des données.

Fichiers de données

Chaque tablespace peut avoir un ou plusieurs *fichiers de données* ; les fichiers physiques réels sont stockés dans le système de fichiers.

Extents

Un *extent* est la plus petite unité de mémoire dans une base de données Oracle. Un extent est constitué d'un seul ou de plusieurs blocs du système de fichiers. Par exemple, le système de fichiers Linux **ext2** contient habituellement 2048 octets par bloc, mais on peut le configurer à 1024, 2048, ou 4096 octets par bloc. Un extent peut être constitué d'un ou de plusieurs blocs **ext2,** selon la manière dont la base de données est utilisée.

© copyright Éditions Eyrolles

Segments

Un segment est la représentation physique d'un objet et se compose d'un ou de plusieurs extents. Une table simple, par exemple, est un segment.

Fichiers de contrôle

Quand la base de données est démarrée, elle se réfère à un fichier binaire, appelé *fichier de contrôle*, pour trouver l'emplacement des fichiers de données et des journaux en ligne. Le fichier de contrôle est mis à jour de manière continue tant que la base de données est active ; il doit être disponible au montage ou à l'ouverture de la base de données. Le fichier de contrôle est habituellement dupliqué sur plusieurs périphériques : si l'un des fichiers de contrôle utilisés par la base de données n'est plus disponible, alors la base de données peut quand même fonctionner correctement.

Journaux

Les journaux stockent tous les changements effectués dans la mémoire cache de la base de données, à quelques exceptions près. Ils servent à minimiser la perte de données, et sont utiles lors de la reconstruction de la base de données. Oracle requiert au moins deux fichiers journaux et des ensembles de fichiers sont souvent mis en miroir dans des périphériques multiples. Leur configuration et leur taille peuvent affecter les performances. Il est courant de stocker les journaux sur des périphériques spécifiques, séparément des fichiers de données. Les journaux sont écrits séquentiellement, à la différence des segments de rollback, qui sont présents dans les fichiers des données, au même titre que les objets de base de données.

Segments de rollback

Les segments de rollback préservent la cohérence en mémorisant les données qui ont été modifiées mais avant validation effectuée dans la base de données : ils sont utilisés pour restaurer les informations après une commande d'annulation de transaction **rollback**. Quand commence une transaction qui modifie les données d'une table, les données initiales sont copiées dans un segment de rollback. Si une seconde transaction a besoin des mêmes données et que la première transaction n'a pas validé les modifications, Oracle va chercher les données du segment de rollback.

© copyright Éditions Eyrolles

Fichiers de paramètres

Ces fichiers spécifient la configuration de l'instance. Le fichier est habituellement appelé **INIT.ORA** ou **INIT<SID>.ORA**. Le nom de l'instance (SID) est associé au nom de fichier.

SID

L'identificateur système représente une instance unique. La variable d'environnement **ORACLE_SID** représente l'instance Oracle par défaut. Tout au long de ce chapitre, on supposera que le SID d'Oracle est **ORCL**. Puisque les SID sont propres à un utilisateur et que certains fichiers utilisent l'identificateur SID comme partie du nom de fichier, on trouvera la notation <SID> dans la documentation. En principe, le nom d'instance a quatre caractères ou moins.

$ORACLE_HOME

La variable d'environnement **ORACLE_HOME** sert à identifier une installation particulière d'Oracle, sachant que l'on peut installer plusieurs versions d'Oracle. **ORACLE_HOME** peut être modifiée pour pointer vers différentes versions. Dans ce document, la notation **$ORACLE_HOME** représentera la structure du répertoire réel dans lequel Oracle est installé. On utilise habituellement les répertoires :

- /u01/app/oracle/product/8.0.5 pour Oracle8,
- /u01/app/oracle/product/8.1.5 pour Oracle8i.

$ORACLE_BASE

La variable d'environnement **ORACLE_BASE** sert à identifier la base de l'arbre d'installation d'Oracle.

Identifiants de version

Oracle a un identificateur de version à cinq chiffres. Le premier chiffre est la version majeure, le second la version mineure, le troisième est un code contenant le groupe d'ingénierie et les quatrième et cinquième chiffres se rapportent à un ensemble de correctifs (*patches*). Les deux ou trois premiers chiffres déterminent un produit individualisé. Par exemple, Oracle8i est la version 8.1.5. Un ensemble de correctifs relatifs à Oracle8i peut porter le numéro de version 8.1.5.0.2. Le chiffre le plus à droite est aussi le moins significatif par rapport au changement de version.

© copyright Éditions Eyrolles

PL/SQL

PL/SQL est un langage procédural d'Oracle, qui ajoute à SQL la logique d'application, dont des constructions de procédures par blocs structurés. Il sert à écrire des applications qui manipulent les données dans la base.

Schéma

Le schéma représente tous les objets possédés par un compte particulier.

Installation

L'installation d'Oracle exige une planification rigoureuse ; il faut aussi être attentif aux détails. Cette section comprend des instructions valables pour l'installation de Oracle8 et Oracle8i. Bien qu'elles soient similaires, elles présentent quelques différences importantes. La section est divisée en quatre parties : pré-installation, installation d'Oracle8, installation d'Oracle8i et post-installation. Oracle exige une opulence de ressources matérielles. Généralement, Oracle s'exécute bien mieux avec plus de RAM, ce qui signifie que plus il peut disposer de mémoire pour le rangement de données, moins il a recours aux lecteurs physiques. Oracle fonctionne aussi beaucoup mieux avec des lecteurs en nombre important, parce qu'il y a moins de contention entre les fichiers. La meilleure méthode pour améliorer les performances est d'ajouter de la RAM. La seconde est d'augmenter le nombre de disques. Les bases de données relationnelles sont souvent tributaires des entrées/sorties.

Même si Oracle demande seulement 400 Mo d'espace disque et 32 Mo de RAM au minimum pour Oracle8 (voir tableau 3.1), il vaut mieux prévoir en réalité au moins 500 Mo d'espace disque et 128 Mo de RAM.

Tableau 3–1
Caractéristiques minimales pour utiliser Oracle8

Caractéristiques	Valeur minimale
Espace disque	400 Mo
Mémoire	32 Mo RAM
Espace d'échange	3 fois la valeur de la RAM

© copyright Éditions Eyrolles

Tableau 3–1
Caractéristiques minimales pour utiliser Oracle8

Caractéristiques	Valeur minimale
Système d'exploitation	Noyau Linux 2.0.34 ou supérieur
Bibliothèques système	Bibliothèque GNU C version 2.0.7

Si vous installez Oracle8i (voir tableau 3.2), il faut en réalité au moins 128 Mo de RAM pour prendre en charge l'environnement Run-Time Java, et un minimum de 198 Mo pour un petit environnement de production. Il faudra 1000 Mo d'espace disque pour recevoir et faire fonctionner le logiciel et les fichiers de la base de données.

Tableau 3–2
Caractéristiques minimales pour utiliser Oracle8i

Caractéristiques	Valeur minimale
Espace disque	693 Mo
Mémoire	128 Mo RAM
Espace d'échange	3 fois la valeur de RAM
Système d'exploitation	Noyau Linux 2.2
Bibliothèques système	Bibliothèque GNU C version 2.1

Pré-installation

Il faut préparer Linux à l'installation du SGBDR Oracle. Ce n'est pas une application que l'on peut installer dans la minute où l'on reçoit le CD-ROM ou dès que l'on a téléchargé le logiciel. Cela demande à la fois temps et patience. La documentation du logiciel Oracle est accessible à l'adresse http://technet.oracle.com/docs/index.htm. Avant d'installer Oracle, il est vivement conseillé de lire la *Note de version d'Oracle pour LINUX* et le *Guide d'installation pour LINUX*. Les instructions de pré-installation pour Oracle8 et Oracle8i sont assez similaires, sauf que Oracle8i a besoin de Java et d'un **$ORACLE_HOME** différent. Suivre les étapes de pré-installation ci-dessous :

1. Configurer le système de fichiers.
2. Configurer l'espace d'échange.
3. Initialiser les paramètres du noyau.

© copyright Éditions Eyrolles

4. Créer les comptes d'utilisateur et les groupes.
5. Initialiser l'environnement Linux.
6. Installer les paquetages Linux nécessaires.
7. Installer Java.
8. Obtenir les logiciels Oracle et les correctifs.

Configurer le système de fichiers

Les points de montage sont les premiers éléments à prendre en considération. L'usage du disque est souvent plus facile à déterminer en phase de conception et de construction d'un serveur. Le minimum, pour un système de production, sera de six lecteurs ; pour un système de développement, de quatre lecteurs. Des disques peuvent toujours être ajoutés si nécessaire. Le point le plus important concerne la sécurité des données : les fichiers de données et les journaux résideront sur des périphériques séparés. Si l'un des lecteurs contenant des données est perdu, les journaux peuvent être utilisés pour restaurer les données. Généralement, il est préférable de disposer d'un grand nombre de lecteurs de capacité relativement petite que d'un nombre réduit de lecteurs de grande capacité. Si l'on a le choix entre six lecteurs de 2 Go et deux lecteurs de 9 Go, choisir la configuration à six disques ; augmenter le nombre de lecteurs apporte de meilleurs performances et une meilleur protection des données. Un autre point important à considérer est la contention des lecteurs entre les différentes applications. Un lecteur de 9 Go réparti en deux partitions de 4,5 Go, partagées entre deux applications différentes, sera difficile à optimiser. Quand c'est possible, il vaut mieux dédier les lecteurs à Oracle.

Oracle recommande une organisation que l'on appelle une architecture flexible optimale, ou OFA (*Optimal Flexible Architecture*). OFA est une méthode particulière de disposition des systèmes de fichiers, qui exige un minimum de quatre points de montage situés au même niveau de la structure du répertoire. C'est particulièrement utile si l'on exécute des instances multiples de bases de données sur le même serveur. La documentation Oracle se reporte souvent à l'OFA, aussi est-elle très utile si l'on se sert de cette architecture.

Un seul disque

On peut installer Oracle sur un lecteur physique unique, s'il y a peu de risques en termes de performances ou de pertes de données. Le dispositif que je vais décrire peut sembler étrange et inutile, mais il vous donnera l'habitude de ranger les données dans les partitions, même s'il n'est pas fondé sur des lecteurs physiques. On se souviendra bien que cela n'est pas recommandé dans un environnement de production, car on risque de perdre des données et d'avoir des performances très faibles.

© copyright Éditions Eyrolles

Localiser sur votre système une partition disposant d'au moins 600 Mo d'espace libre à l'aide de la commande **df –k.** Supposons que la partition **/usr/** ait 1 Go d'espace disque disponible. Créer un répertoire sous **Oracle** appelé **/usr/local**, puis créer quatre répertoires sous **Oracle** appelés **u01**, **u02**, **u03** et **u04**.

```
$ df -k
Filesystem 1k-blocks Used Available Use% Mounted on
/dev/sda3 3865709 2663841 1201868 69% /usr

mkdir /usr/local/oracle
mkdir /usr/local/oracle/u01
mkdir /usr/local/oracle/u02
mkdir /usr/local/oracle/u03
mkdir /usr/local/oracle/u04
```

Les exemples, dans la suite du chapitre, supposent que les points de montage sont situés sous la racine **/** du système de fichiers. Créer des liens symboliques sous la racine du système de fichiers.

```
ln –s /usr/local/oracle/u01 /u01
ln –s /usr/local/oracle/u02 /u02
ln –s /usr/local/oracle/u03 /u03
ln –s /usr/local/oracle/u04 /u04
```

Quatre disques

Une configuration à quatre disques accroît sensiblement la protection des données et améliore quelque peu les performances, par rapport à une configuration mono-disque : créer une partition sur chaque lecteur physique pour ranger les données Oracle ; assigner une partition à chaque répertoire Oracle, **/u01**, **/u02**, **/u03**, et **/u04** ; si Linux est déjà installé, et que l'on veut ajouter trois disques au système, utiliser Disk Druid ou **fdisk** pour réaliser la partition des disques et **mkfs** pour formater les partitions ; créer les répertoires **/u01**, **/u02**, **/u03** et **/u04** sous la racine **/** du système de fichier et utiliser la commande **mount** pour monter les partitions dans le système de fichiers.

Configurer l'espace d'échange

La taille de l'espace d'échange sera de trois fois la taille de la RAM. Red Hat Linux 6.0 et le noyau Linux 2.2 supportent maintenant des partitions d'échange de plus de 128 Mo. Pour créer des partitions d'échange ou des fichiers d'échange plus grands que 128 Mo, spécifier le nouveau style, **mkswap –v1**. En l'absence d'espace disque disponible pour ajouter une partition d'échange, créer un fichier d'échange.

© copyright Éditions Eyrolles

Paramètres du noyau

Le *Guide d'installation Oracle pour LINUX* recommande de changer les paramètres du noyau, ce qui exige une recompilation du noyau. Normalement, il n'est pas nécessaire de modifier ces paramètres. Le paramètre le plus important, **SHMMAX**, définit la taille maximale d'un segment de mémoire partagée : l'option par défaut est de 32 Mo sur Red Hat Linux, ce qui est généralement suffisant pour l'installation d'une base de données de premier niveau. Si la SGA occupe 128 Mo, il faudra quatre segments de 32 Mo chacun. Les paramètres du noyau posent plus de problèmes sur un serveur doté d'une grande SGA. Dans un environnement de production, **SHMMAX** aura la même taille que la RAM physique. Si vous décidez de faire ces modifications, les fichiers à modifier sont usr/src/linux/include/asm/shmparam.h et /usr/src/linux/include/linux/sem.h.

Remarque. On obtient la valeur courante du paramètre SHMMAX avec la commande suivante, passée en tant qu'utilisateur root : **cat /proc/sys/kernel/shmmax**.

Comptes et groupes système

Le serveur Oracle ne s'exécutera pas sous le compte root. On a l'habitude de créer un compte, appelé **Oracle**, propriétaire du logiciel serveur Oracle et du processus en tâche de fond. Oracle définit un groupe à l'intérieur du serveur Oracle, appelé OSDBA. Tout compte Oracle membre d'OSDBA reçoit l'autorisation DBA. Oracle se sert d'un groupe Linux, **/etc/group**, pour assigner la qualité de membre au groupe OSDBA du serveur Oracle. À l'installation, Oracle a besoin d'un groupe Linux pour définir la qualité de membre de l'OSDBA — c'est souvent le groupe Linux **dba** qui en fait office. Oracle8i dispose d'un autre groupe, **oinstall**, utilisé pour affecter l'autorisation d'installer le logiciel dans le référentiel.

Utiliser, en tant que root, la commande **groupadd** pour ajouter le groupe **dba** :

```
# groupadd dba
```

Pour Oracle8i, ajouter aussi le groupe **oinstall** :

```
# groupadd oinstall
```

Pour Oracle8, créer le compte Oracle en se servant du groupe **dba** en tant que groupe par défaut :

```
# useradd –c DBA –g dba oracle
```

© copyright Éditions Eyrolles

Pour Oracle8i, créer le compte Oracle en se servant du groupe **oinstall** en tant que groupe par défaut :

```
# useradd –c DBA –g oinstall –G dba oracle
```

Créer un mot de passe pour le compte Oracle :

```
# passwd oracle
```

Droits sur le système de fichiers

Pour que le compte **Oracle** puisse utiliser les points de montage pendant l'installation, il doit avoir les droits appropriés : il s'agit de donner la propriété des points de montage au compte **Oracle.** En fin d'installation, loin rend la propriété des répertoires principaux à root.

```
# chown oracle.dba /u01
# chown oracle.dba /u02
# chown oracle.dba /u03
# chown oracle.dba /u04
# chmod 755 /u01
# chmod 755 /u02
# chmod 755 /u03
# chmod 755 /u04
```

Options d'environnement

Certaines variables d'environnement doivent être initialisées avant que le serveur Oracle ne soit installé. Il existe trois endroits où placer les variables d'environnement : si le compte **oracle** est seul à se servir du logiciel Oracle, placer les variables d'environnement dans le fichier **~/.bash_profile** du compte Oracle ; si d'autres comptes sur le système sont appelés à se servir d'Oracle, les variables peuvent être placées dans le fichier système **/etc/profile**. L'inconvénient de cet emplacement est qu'il faut avoir la qualité root pour modifier les options de **/etc/profile**. Enfin, si des bases de données multiples sont installées sur le serveur, il vaut mieux placer les variables à part, dans un script exécutable par l'utilisateur. Oracle utilise un fichier appelé **oraenv** situé sous **$ORACLE_HOME/bin**.

Du fait que les variables d'environnement doivent être initialisées avant l'installation, il faut insérer les variables d'environnement suivantes, soit dans le fichier **~/.bash_profile** du compte Oracle, soit dans **/etc/profile**, et se reconnecter pour initialiser l'environnement ou charger le fichier. Remarquez que quelques-uns des répertoires cités dans les options d'environnement n'existent pas encore à ce niveau. Dans l'exemple suivant, l'instance **ORACLE_SID** est affectée à **orcl**.

© copyright Éditions Eyrolles

Options d'environnement pour Oracle8 (8.0.5) :

```
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=/u01/app/oracle/product/8.0.5
ORACLE_SID=orcl
ORACLE_TERM=vt100
PATH=$PATH:$ORACLE_HOME/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ORACLE_HOME/lib:
    $ORACLE_HOME/network/lib
export ORACLE_BASE ORACLE_HOME ORACLE_SID ORACLE_TERM
LD_LIBRARY_PATH PATH
```

Options d'environnement pour Oracle8i (8.1.5) :

```
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=/u01/app/oracle/product/8.1.5
ORACLE_SID=orcl
ORACLE_TERM=vt100
PATH=$PATH:$ORACLE_HOME/bin
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ORACLE_HOME/lib
    :$ORACLE_HOME/network/lib
export ORACLE_BASE ORACLE_HOME ORACLE_SID ORACLE_TERM
LD_LIBRARY_PATH PATH
```

Important. La variable d'environnement **LD_LIBRARY_PATH** gère l'emplacement des bibliothèques relatives aux applications. Certaines personnes ont rencontré des problèmes car Oracle ne trouve pas certaines bibliothèques dans le **LD_LIBRARY_PATH**. Les répertoires contenant les bibliothèques peuvent être ajoutés au fichier **/etc/ld.so.conf.** La commande **ldconfig** relira le fichier **ld.so.conf.**

Si vous avez besoin d'une langue autre que la langue par défaut, US7ASCII, vérifier la documentation d'installation d'Oracle au sujet des variables **NLS_LANG** et **ORA_NLS33.** Copier ces variables d'environnement dans votre fichier de démarrage du shell.

Si vous souhaiter aller dans le répertoire **$ORACLE_HOME**, le raccourci à utiliser est **cd $ORACLE_HOME**. Une méthode plus facile est de créer un alias : placer la commande suivante dans le répertoire **/etc/bashrc** ; une fois l'alias défini, la commande **oh** vous amènera dans **ORACLE_HOME**.

```
alias oh='cd $ORACLE_HOME'
```

Exécuter la commande **umask** pour vous assurer que le masque de création de fichier utilisateur (umask) est 022. Si l'umask ne vaut pas 022, placer **umask 022** dans votre fichier de démarrage du shell. Ce ne sera pas nécessaire dans la plupart des distributions de Linux, dont Red Hat : l'umask est un filtre des droits sur le système de fichiers. Un umask de 022 donne tous les droits au propriétaire, et enlève le droit d'écriture au groupe et à tous les autres.

© copyright Éditions Eyrolles

```
$ umask
022
```

Avant de continuer, exécuter la commande **env** pour s'assurer que les variables d'environnement ont été configurées correctement.

Affichage X Window

Le serveur X Window n'autorise pas d'autres processus utilisateur à accéder à l'écran par défaut. Si vous installez Oracle8i, les utilitaires graphiques Java ont besoin d'un accès au serveur X Window. Tant que vous êtes connecté sous le compte utilisateur **oracle** et que vous exécutez les applications sous le compte **oracle,** les applications auront accès au serveur X. Si vous êtes connecté en tant que root, que vous passez au compte **oracle** avec la commande **su** et que vous tentez d'exécuter une application dans une fenêtre X en tant que **oracle**, vous pouvez être mis en échec. Pour l'éviter, sous le compte utilisateur avec lequel vous vous êtes connecté, exécutez la commande **xhost +localhost** : la commande **xhost** permet à des comptes divers d'accéder au serveur X. Ensuite, sous l'autre compte utilisateur, initialisez la variable d'environnement **DISPLAY** à **localhost**.

```
$ xhost +localhost
$ export DISPLAY=localhost:0.0
```

Paquetages Linux

Le passage de la version 5.2 de Red Hat Linux à la version 6.0 a introduit des changements fondamentaux dans les bibliothèques du noyau. Red Hat 5.2 comprend les bibliothèques **glibc** en version 2.0, et Red Hat Linux 6.0 les nouvelles bibliothèques en version 2.1 de **glibc**. Les applications compilées avec **glibc** 2.0 doivent être recompilées avec les nouvelles bibliothèques. Oracle8 a été compilé pour Red Hat Linux 5.2 avec **glibc** 2.0, et exige une compatibilité avec les anciennes bibliothèques pour fonctionner sur les versions plus récentes de Red Hat Linux. Si vous décidez d'installer Oracle8 sur Red Hat 6.0 ou une version ultérieure, vous devez installer les bibliothèques répondant à la compatibilité. Oracle8i a été compilé avec **glibc** 2.1 et par conséquent n'a pas besoin des paquetages de compatibilité, mais Oracle8i ne fonctionnera pas sous les versions de Red Hat Linux antérieures à la version 6.0 parce que les bibliothèques sont incompatibles.

Les paquetages de compatibilité ne sont donc nécessaires que si vous installez Oracle 8.0.5, et vous n'en avez pas besoin si vous installez Oracle 8.1.5 sur Red Hat Linux 6.0 ou 6.1. On trouve les paquetages sur le CD-ROM d'installation de Red Hat Linux ou sur un site ftp.

© copyright Éditions Eyrolles

```
compat-binutils-5.2-2.9.1.0.23.1.i386.rpm
compat-egcs-5.2-1.0.3a.1.i386.rpm
compat-egcs-c++-5.2-1.0.3a.1.i386.rpm
compat-egcs-g77-5.2-1.0.3a.1.i386.rpm
compat-egcs-objc-5.2-1.0.3a.1.i386.rpm
compat-glibc-5.2-2.0.7.1.i386.rpm
compat-libs-5.2-2.i386.rpm
```

Le paquetage de développement C doit être installé. Si vous avez installé Red Hat 6.0 ou 6.1 et sélectionné la station de travail GNOME, la station de travail KDE, une installation de type serveur (*Server-Class*) ou une installation personnalisée incluant le groupe *Development Packages*, le logiciel de développement aura été installé. Pour utiliser l'agent intelligent Oracle (*Oracle Intelligent Agent*), installer TCL 7.5 depuis le site ftp://www.scriptics.com. Si vous ne savez pas ce que c'est, ne l'installez pas.

Java pour Linux

Si l'on installe Oracle8i, installer d'abord un environnement d'exécution Java. L'outil d'installation d'Oracle8 en mode caractère **orainst** a été remplacé dans Oracle8i par un nouveau programme d'installation graphique écrit en Java, appelé programme d'installation universel Oracle (OUI, *Oracle Universal Installer*). Certains ont regretté que le programme d'installation exige une machine exécutant X Window. Si votre système ne dispose pas de X Window, vous pouvez poursuivre votre session X Window sur une autre machine qui exécute un serveur X Window.

Télécharger l'environnement d'exécution Java (JRE, *Java Runtime Environment*) pour Linux de Blackdown sur le site http://www.blackdown.org : il faut télécharger la version **JRE 1.1.6v5** ; choisir la version **libc5** pour Red Hat 5.2, et la version **glibc** pour Red Hat 6.0 et suivantes ; copier le JRE dans le répertoire **/usr/local** ; décompresser les fichiers et créer un lien symbolique vers **/usr/local/jre**, car il semble que le programme d'installation universel s'attende à trouver le JRE sous **/usr/local/jre**.

```
# cd /usr/local
# tar zxvf jre_1.1.6-v5-glibc-x86.tar.gz
# ln -s jre116_v5 jre
```

Placer **/usr/local/jre/bin** dans votre variable d'environnement PATH.

```
PATH=$PATH:/usr/local/jre/bin ; export PATH
```

Attention. Certains problèmes ont été rencontrés avec des versions du JRE autres que la version 1.1.6v5 : par exemple, le programme d'installation qui ne démarre pas, ou certaines étapes de l'installation qui ne se terminent pas. Ne considérez pas qu'une

© copyright Éditions Eyrolles

version plus récente fonctionnera mieux. N'utilisez pas non plus le JDK, pour les mêmes raisons. Si une installation d'Oracle antérieure a échoué, il faut supprimer tous les fichiers précédemment installés avant de réinstaller.

Se procurer le logiciel Oracle

Oracle8 et Oracle8i peuvent être achetés sur le site web Oracle Store à http://oraclestore.oracle.com/ ou être téléchargés depuis le site ftp ftp://ftp.oracle.com/pub/www/otn/linux. On peut aussi souscrire à l'offre *OTN Technology Track for Linux* sur le site OTN (*Oracle Technology Network)* à http://technet.oracle.com. Le service de souscription comprend douze mois de mise à jour. Oracle propose un site dédié à Linux à l'adresse http://platforms.oracle.com/linux/. Si vous décidez de télécharger le logiciel, attendez-vous à ce que ce soit très long. On a le choix entre télécharger un seul grand fichier, ou des fichiers multiples, plus petits.

Correctifs Oracle

Se connecter en tant qu'utilisateur **oracle** et créer un répertoire appelé **patches** ; télécharger les correctifs les plus récents disponibles. Pour Oracle8, télécharger les fichiers depuis ftp://oracle-ftp.oracle.com/server/patchsets/unix/LINUX/80patchsets/80510/ ou l'OTN à http://technet.oracle.com — l'OTN vous demandera l'ouverture d'un compte gratuit. Les correctifs pour Oracle8i se trouvent à l'adresse http://technet.oracle.com/support/tech/linux/support_index.htm.

Placer les correctifs téléchargés dans le répertoire des correctifs : **glibcpatch.tgz**, **lnxpatch.tgz**, et **linux_80510patchset.tgz**. Si vous installez Oracle8 sur Red Hat 5.2, vous n'avez pas besoin de **glibcpatch.tgz**. Pour Oracle8i, il faut aussi le correctif **linux_815patches.tgz**. Lisez bien le README relatif à chaque correctif ; d'autre part, il y aura aussi des FAQ pour donner des pistes ou des critiques utiles. Si vous rencontrez des problèmes, ce sont les premières choses à consulter.

Installer Oracle8

Oracle8 s'appuie sur un programme d'installation appelé *Software Asset Manager*, ou **orainst**. Le programme d'installation est parfois peu commode. Utiliser la touche **TAB** pour se déplacer entre les champs et boutons, la barre d'espacement pour choisir une option et la touche **ENTER** pour activer un bouton. Si un champ est trop court (à l'écran) pour afficher son contenu complet, sélectionner le champ à l'aide de la touche **TAB,** et utiliser la touche flèche droite pour afficher la suite du texte.

© copyright Éditions Eyrolles

1. Se connecter sous le compte **oracle**.
2. Passer dans le répertoire contenant le programme d'installation Oracle.

```
cd /mnt/cdrom/orainst
```

3. Démarrer le programme d'installation Oracle en mode texte. (Le mode graphique Motif ne fonctionnera pas sous Linux). Voir la figure 3.1.

```
./orainst /c
```

Remarque. Si l'on se sert d'un CD-ROM et que le script **orainst** ne s'exécute pas, s'assurer que le CD-ROM est bien monté avec le paramètre exec.

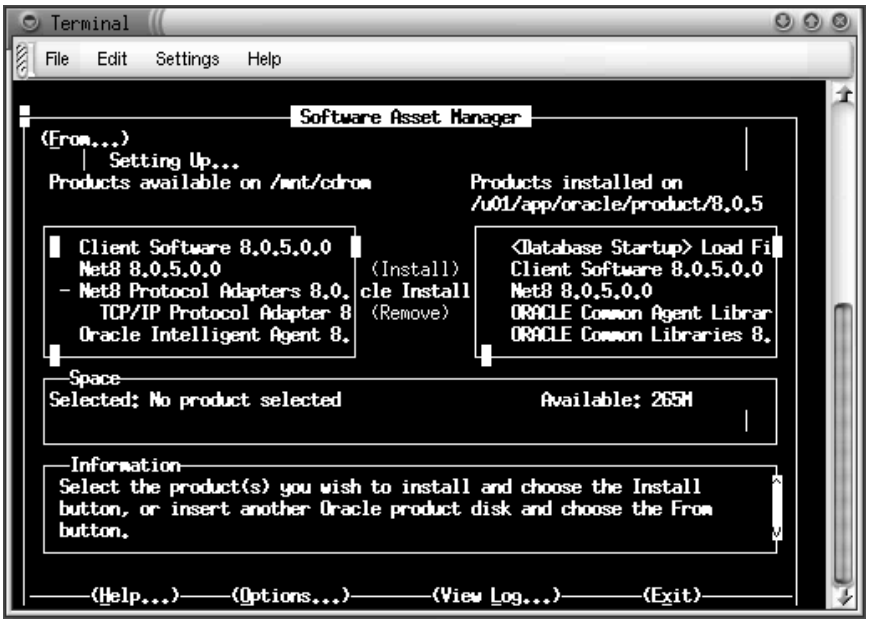

Figure 3–1
Programme d'installation d'Oracle8

4. Choisir **Custom Install**. Appuyer sur la touche **TAB** pour passer à la boîte de sélection du type d'installation, utiliser les flèches pour choisir **Custom**, et appuyer sur la touche **ENTER** pour continuer.
5. Entrer **OK** pour afficher le texte de README.
6. Choisir **Install**, **Upgrade** ou **De-install software**.
7. Choisir **Install new product – Do Not Create DB objects**.
8. Vérifier **ORACLE_HOME** et **ORACLE_BASE**.

© copyright Éditions Eyrolles

9. Accepter les options par défaut pour les fichiers de trace (log).
10. Choisir **Install from CD-ROM**.
11. Choisir une langue.
12. Le programme d'installation explique ensuite où le script **root.sh** sera placé pour un usage ultérieur.
13. L'écran des options d'installation apparaît. Pour voir une description complète d'une option, placer le curseur sur l'option à l'aide de la touche **TAB,** puis appuyer sur la flèche droite du clavier ; appuyer sur la barre d'espacement pour choisir une option ; ne pas choisir la documentation du serveur Oracle8 parce qu'il y a une erreur dans la structure du répertoire qui met l'installation en échec ; ne pas choisir les pilotes JDBC ou l'agent intelligent à ce moment de l'installation. Les options à retenir sont les suivantes : Client Software, Net8, TCP/IP Protocol Adapter, Oracle Unix Installer, la version Standard ou Entreprise d'Oracle8, PL/SQL, Pro*C/C++ et SQL*Plus.
14. Quand la sélection est terminée, choisir le bouton **Install** et appuyer sur la touche **ENTER**.
15. Entrer **OK** à la vue du message **ulimit.**
16. Choisir **dba** comme groupe **dba.**
17. Entrer **OK** sur l'invite OSOPER.
18. L'installation commence. Une fois achevée, un message indique « The requested action has been performed for the selected products » — L'action requise a été exécutée pour les produits choisis.
19. Un message apparaît, qui demande le compte root pour exécuter le script **root.sh.** Entrer **OK**.
20. Sortir du programme d'installation.
21. Ouvrir une nouvelle fenêtre terminal et passer au compte root avec **su**. En tant que root, aller dans le répertoire **$ORACLE_HOME/orainst** et exécuter la commande **sh root.sh**. Vérifier les variables d'environnement **ORACLE_OWNER**, **ORACLE_HOME** et **ORACLE_SID**. Si les variables ne sont pas correctes, sortir du script et initialiser les variables d'environnement avec les valeurs fournies précédemment dans ce chapitre. Le nom complet du chemin du répertoire **bin** local est **/usr/local/bin**. Ne pas tenir compte de l'indication que **ORACLE_HOME** ne concorde pas avec le répertoire home du compte **oracle**.

© copyright Éditions Eyrolles

```
[root@dbsvr orainst]# cd $ORACLE_HOME/orainst
[root@dbsvr orainst]# sh root.sh

Running Oracle8 root.sh script...

The following environment variables are set as:
   ORACLE_OWNER= oracle
   ORACLE_HOME= /u01/app/oracle/product/8.0.5
   ORACLE_SID= orcl
Are these settings correct (Y/N)? [Y]:

Enter the full pathname of the local bin directory [/usr/lbin]: /usr/local/bin

Checking for "oracle" user id...
ORACLE_HOME does not match the home directory for oracle.
Okay to continue? [N]: Y
Replacing existing entry for SID in /etc/oratab...
The previous entry is now in a comment.

Leaving common section of Oracle8 root.sh.
```

Installer les correctifs

Quand Oracle8 est installé, il faut installer les correctifs. Si vous ne l'avez pas encore fait, téléchargez-les à partir de ftp://oracle-ftp.oracle.com/server/patchsets/unix/LINUX/80patchsets/80510/. Si vous avez besoin du correctif **glibcpatch.tgz** et que vous ne le trouvez pas sur le site précédemment cité, ouvrez un compte libre sur le site de l'OTN (http://technet.oracle.com) et cherchez la section des logiciels pour Linux. Recherchez les nouvelles versions des correctifs. Actuellement, il existe trois correctifs pour Oracle8 : **glibcpatch.tgz**, **lnxpatch.tgz**, et **linux_80510patchset.tgz**. **glibcpatch.tgz** n'est utile que si vous installez Oracle 8.0.5 sur Red Hat Linux version 6.0 ou ultérieure. Il faudra peut-être réinstaller le **glibcpatch** après application de nouveaux correctifs. Si vous installez une machine Red Hat 5.2, ce n'est pas la peine : le fichier **lnxpatch.tgz** comprend des correctifs spécifiques à la plate-forme Linux. Enfin, **linux_80510patchset.tgz** contient des corrections génériques pour Oracle8. Dans tous les cas, lire toute la documentation fournie avec les correctifs.

Lnxpatch

Le correctif **lnxpatch** est un script de shell.

1. Aller dans le répertoire contenant les correctifs.
2. Créer un répertoire appelé **lnx**.

© copyright Éditions Eyrolles

3. Transférer le fichier **lnxpatch.tgz** dans le répertoire **lnx**.

```
mv lnxpatch.tgz lnx
```

4. Aller dans le répertoire **lnx**.

```
cd lnx
```

5. Extraire le correctif.

```
tar zxvf lnxpatch.tgz
```

6. Exécuter le script **lnxpatch.sh**.

```
sh glibcpatch.sh
```

80510patchset

Le fichier 80510patchset doit être installé par le programme d'installation Oracle, **orainst**, que vous avez utilisé pour installer le serveur Oracle.

1. Créer un répertoire appelé **80510**.
2. Transférer le fichier **80510patchset.tgz** dans le répertoire **80510**.

```
mv 80510patchset.tgz 80510
```

3. Aller dans le répertoire **80510**.

```
cd 80510
```

4. Extraire le correctif.

```
tar zxvf 80510patchset.tgz
```

5. Démarrer le programme d'installation Oracle.

```
orainst /c
```

6. Choisir **Custom Install**.
7. Entrer **OK** pour afficher le README.
8. Choisir **Install, Upgrade, or De-install Software**.
9. Choisir **Add/Upgrade Software**.
10. Vérifier le répertoire **ORACLE_HOME.**
11. Confirmer les emplacements des fichiers de trace et entrer **OK**.
12. Choisir **Install from Staging Area**.

© copyright Éditions Eyrolles

13. Entrer le chemin complet du répertoire où le correctif **8051** est situé ; par exemple, **/home/oracle/patches/8.0.5/8051/**.
14. Choisir une langue.
15. Créer un nouveau fichier **root.sh.**
16. Choisir **OK** pour sauvegarder le fichier **root.sh** précédent.
17. Choisir **OK** au vu du message de l'étape de post-installation.
18. Choisir les éléments listés dans la boîte de gauche, Oracle8 Standard et Precomp.
19. Choisir **Install**.
20. Entrer **Yes** pour mettre à jour chaque composant.
21. Entrer **OK** au vu du message ulimit.
22. Choisir **dba** en tant que groupe et entrer **OK**.
23. Entrer **OK** pour le groupe OSOPER.
24. Une fois l'installation terminée, sortir du programme d'installation.
25. Exécuter le script **root.sh** en tant que root.

Glibcpatch

Le correctif **glibcpatch** est un script de shell.

1. Aller dans le répertoire contenant les correctifs.
2. Créer un répertoire appelé **glibc**
3. Transférer le fichier **glibcpatch.tgz** dans le répertoire **glibc**.

```
mv glibcpatch.tgz glibc
```

4. Aller dans le répertoire **glibc**.

```
cd glibc
```

5. Extraire le correctif.

```
tar zxvf glibcpatch.tgz
```

6. Exécuter le script **glibcpatch**.

```
sh glibcpatch.sh
```

© copyright Éditions Eyrolles

Créer les objets de base de données

Suivre exactement ces étapes pour créer les objets d'une base de données.

1. Aller dans le répertoire contenant le logiciel d'installation Oracle.

```
cd /mnt/cdrom/orainst
```

2. Démarrer le programme d'installation Oracle.

```
./orainst /c
```

3. Choisir **Custom Install**.
4. Entrer **OK** au vu du prompt README.
5. Choisir **Create/Upgrade Database Objects**.
6. Choisir **Create Database Objects**.
7. Vérifier **ORACLE_HOME** et **ORACLE_BASE**.
8. Accepter les options par défaut pour les fichiers de trace.
9. Vérifier que **ORACLE_SID** est correct.
10. Si un message indique que le SID est déjà en cours d'utilisation, il est dû au fait que le script **root.sh** a créé un fichier **/etc/oratab**. Entrer **Yes**.
11. Dans l'écran d'installation d'Oracle, choisir **Oracle8 Standard RDBMS 8.0.5.0.0** pour créer les objets de base de données.
12. Cliquer sur le bouton **Install**.
13. Choisir **Create Product DB objects**.
14. Choisir **Filesystem-based Database**.
15. Si vous initialisez un système de fichiers compatible OFA (*Optimal Flexible Architecture*), choisir **Yes** pour avoir une distribution des fichiers de contrôle sur des points de montage multiples.
16. Entrer les points de montage **/u02**, **/u03**, et **/u04** pour initialiser le système de fichiers OFA.
17. Choisir le jeu de caractères.
18. Choisir le jeu de caractères national.
19. Entrer votre mot de passe pour le compte SYS.
20. Entrer votre mot de passe pour le compte SYSTEM.
21. Entrer éventuellement un mot de passe interne pour le dba et l'opérateur.

© copyright Éditions Eyrolles

22. Entrer un mot de passe pour le TNS Listener.
23. Ne pas configurer le MTS listener.
24. Accepter les options par défaut pour les fichiers de contrôle.
25. Choisir **OK** dans les deux écrans suivants.
26. Choix **Yes** pour accepter les options par défaut.
27. Le processus de construction de la base de données commence, le serveur Oracle est démarré, et les objets sont créés.
28. Sortir du programme d'installation.

Installer Oracle8i

L'installation d'Oracle8i est délicate parce qu'il faut louvoyer entre les bogues du sous-programme d'installation pour la version 8.1.5. Le but est d'installer d'abord le logiciel Oracle, puis d'installer les correctifs, de créer une base de données réelle appelée **orcl** et de démarrer l'instance.

Les instructions d'installation qui suivent ont été exécutées sur un système équipé d'une copie neuve de Red Hat Linux 6.1, incluant tous les correctifs disponibles. Il faut s'attendre à ce que le programme d'installation universel Oracle (OUI, *Oracle Universal Installer*) achoppe sur un problème de mémoire. Il faut au moins 128 Mo de RAM, et un espace d'échange suffisant pour commencer l'installation. S'assurer que le JRE Java est installé et que le lien symbolique **/usr/local/jre** existe ; que vous avez un compte **Oracle,** un groupe **dba** et un groupe **oinstaller —** Oracle sera membre des deux groupes, **oinstall** étant le groupe primaire listé dans **/etc/passwd —** ; que les variables d'environnement Oracle existent, à l'aide de la commande **env.**

Attention. Il y a eu des problèmes à l'installation sur les versions par défaut de Enlightenment installé avec Red Hat Linux 6.0. Enlightenment est le gestionnaire de fenêtres par défaut installé sous Gnome. Si vous rencontrez des problèmes avec le programme d'installation universel Oracle, essayez de mettre à jour Enlightenment ou remplacez-le par Sawmill. On peut aussi faire appel à **fvwm** ou à KDE.

Se connecter en tant qu'utilisateur **Oracle** pour lancer l'installation, et monter le CD-ROM contenant le logiciel. Le point de montage du CD-ROM est en principe **/mnt/cdrom**. Si le logiciel est téléchargé, remplacer **/mnt/cdrom** par le répertoire où il est installé.

1. Aller dans le répertoire d'installation du CD-ROM Oracle.

```
$ cd /mnt/cdrom/
```

© copyright Éditions Eyrolles

2. Le *Guide d'installation Oracle8i* pour Linux stipule qu'il faut exécuter le script **runInstaller,** mais celui-ci ne fonctionne pas sous Red Hat Linux 6.1. Utiliser le script **runIns.sh** situé dans le répertoire **/mnt/cdrom/install/linux**.

```
$ cd /mnt/cdrom/install/linux

$. /runIns.sh
```

   Si un message indique que l'OUI ne peut pas se connecter à un serveur X11 Window, il est probable que la connexion s'est faite dans le gestionnaire de fenêtre sous un autre compte utilisateur. Vous êtes sans doute passé au compte Oracle par une commande **su** (*setuser*). Reportez-vous à la section X-Window des options d'environnement. S'il se produit une erreur Java bizarre, assurez-vous que vous avez installé la version correcte de Java et que vous avez bien les répertoires **/usr/local/jre** et **usr/local/jre/bin** dans votre $PATH. Utilisez la commande **witch jre** pour vérifier qu'elle trouve bien le répertoire **/usr/local/jre/bin/jre**.

3. OUI propose une boîte de dialogue d'accueil (figure 3.2).

Figure 3–2
Programme d'installation universel Oracle

© copyright Éditions Eyrolles

Remarquer le bouton d'aide situé en bas de la fenêtre OUI. Pendant l'installation, on a des informations utiles en ouvrant la fenêtre d'aide. Cliquer sur **Next**.

4. La boîte de dialogue Files Location attend des précisions sur l'emplacement de la source et de la destination du logiciel. La source sera le CD-ROM ou le répertoire où est placé le logiciel Oracle : **/mnt/cdrom/stage/products.jar** ; la destination sera le répertoire du logiciel OFA : **/u01/app/oracle/product/8.1.5**. Vous pouvez adapter ces répertoires à votre cas mais, si vous avez suivi les indications de ce chapitre, les options par défaut sont correctes. Cliquer sur **Next**.

5. La boîte de dialogue Unix Group Name demande un nom de groupe Unix pour gérer les installations du logiciel. On peut ne pas renseigner cette zone si l'on préfère installer le logiciel Oracle sous le compte root. Entrer le compte **oinstall,** créé auparavant. L'idée est que n'importe quel membre du groupe **oinstall** peut s'occuper du logiciel Oracle, ce qui libère le titulaire du compte root de l'administration d'Oracle. Cliquer sur **Next**.

Oracle Universal Installer
Certain actions need to be performed with root privileges before the install can continue. These actions are stored in a Bourne Shell script called /tmp/orainstall/orainstRoot.sh.
Please execute the /tmp/orainstall/orainstRoot.sh script now and then press "Retry" to continue the install.
OR
Press "Exit", execute the /tmp/orainstall/orainstRoot.sh script, and then restart the installer.
Help...
Retry
Exit

Figure 3–3
Lancement du script orainstRootsh par OUI

6. La première fois que Oracle8i est installé sur un système, apparaît la boîte de dialogue de la figure 3.3 : elle demande le compte root pour exécuter un script appelé **/tmp/OraInstall/orainstRoot.sh**. Ce script crée, dans le répertoire **/etc**, un fichier, appelé **oraInst.loc**, qui contient l'emplacement du répertoire **oraInventory,** et change le propriétaire du groupe du répertoire **oraInventory** en **oinstall**. Si ce dialogue n'apparaît pas, soit vous avez déjà installé Oracle, soit le compte Oracle n'est pas membre du groupe **dba.**

© copyright Éditions Eyrolles

Ouvrir une nouvelle fenêtre terminal et passer au compte root avec la commande **su**. Se mettre, en tant que root, dans le répertoire **/tmp/OraInstall** et entrer **./orainstRoot.sh**. Quand le script **orainstRoot.sh** est terminé, choisir **Retry** dans la boîte de dialogue.

```
# cd /tmp/orainstall
# /orainstRoot.sh
```

7. La boîte de dialogue Avalaible Products (produits disponibles) propose trois options : serveur, client et programmeur. Puisque nous installons le serveur, assurez-vous que l'édition Orace8i Enterprise (ou l'édition standard) est sélectionnée, puis cliquez sur **Next**.

8. La boîte de dialogue Installation Types propose trois options : Typical, Minimal et Custom. Choisir **Custom**, puis cliquer sur **Next**. Les options typiques et minimales ne sont pas censées fonctionner sous Red Hat 6.1.

9. La boîte de dialogue Avalaible Products Components apparaît. S'assurer que toutes les options du produit Oracle sont sélectionnées, sauf interMedia : il y a un bogue dans les versions 8.1.5.0.1 et 8.1.5.0.2 qui pose des problèmes si toutes les options ne sont pas installées. Et, justement, interMedia n'est pas complètement supporté dans la version 8.1.5. Les options produits ne sont pas vraiment utilisées tant que l'on ne les implémente pas. Cliquer sur **Next**.

10. La boîte de dialogue Create Database demande si l'on veut créer une nouvelle base de données à l'aide de l'assistant de configuration de base de données Oracle. Choisir **No** et cliquer sur **Next**.

11. La boîte de dialogue Oracle Protocol Support apparaîtra avec une boîte de sélection vide : c'est normal, car Linux supporte TCP/IP, installé par défaut. Cliquer sur **Next**.

12. La boîte de dialogue Summary apparaît ; choisir **Install** pour commencer l'installation.

13. La boîte de dialogue Setup Privileges apparaîtra si c'est la première fois que l'on installe le logiciel Oracle sur la machine (figure 3.4). Pendant l'installation, l'OUI vérifie qu'il existe un fichier **/etc/oratab** contenant une entrée pour chaque base de données installée. Si le fichier n'existe pas, une boîte de dialogue apparaîtra avec un message qui signifie :
« Exécuter s'il vous plaît le script root.sh situé dans le répertoire /u01/app/oracle/product/8.1.5/root.sh, en vous connectant en tant que root dans une autre fenêtre avant de continuer ».
Ouvrir une autre fenêtre terminal, passer au compte root (**su**), et exécuter le script **root.sh.** Le script demande le nom complet du chemin du répertoire **bin** local : indiquez l'emplacement **/usr/local/bin**. Si, par hasard, vous laissez **/usr/bin** en tant que répertoire **bin** local, tout fonctionne bien —

© copyright Éditions Eyrolles

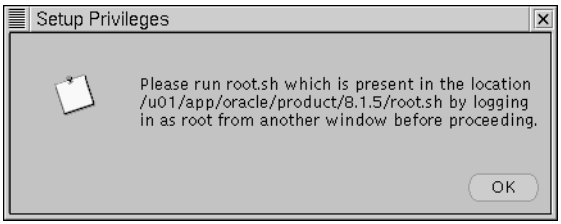
Setup Privileges
Please run root.sh which is present in the location
/u01/app/oracle/product/8.1.5/root.sh by logging
in as root from another window before proceeding.
OK

Figure 3–4
Installer les privilèges

mais ce n'est pas vraiment correct. À la fin, revenir à la fenêtre de dialogue et cliquer sur **OK**.

```
$ su –
Password:
# cd /u01/app/oracle/product/8.1.5/
# sh root.sh
```

14. La boîte de dialogue Configuration Tools apparaît : elle propose l'assistant de configuration de Net8 et les assistants de configuration de la base de données Oracle. Si l'assistant de configuration de base de données Oracle s'arrête, c'est bien. S'il démarre, cliquez simplement sur **Cancel**, puis sur **Yes** dans le dialogue de sortie : la base de données ne doit pas être créée à ce stade parce qu'il faut que les correctifs soient installés auparavant. Fermez tous les messages d'erreur. Vous aurez peut-être un autre message d'erreur de l'OUI ; si c'est la cas, cliquez sur **OK**. Dans la boîte de dialogue des outils de configuration, cliquez sur **Next**.
15. La dernière partie de la boîte de dialogue de fin d'installation apparaît : vous pouvez sortir de l'OUI.
16. Exécuter le script **root.sh** dans le répertoire **$ORACLE_HOME/install/utl** pour modifier quelques permissions de fichier.

```
$ cd $ORACLE_HOME/install/utl
$ sh linux.sh
```

Félicitations, vous avez franchi la première étape de l'installation d'Oracle8i. Maintenant, le logiciel est installé, mais vous n'avez aucune base de données. Avant de créer la base de données, il faut installer les correctifs.

Installer les correctifs

Télécharger tous les correctifs courants pour Oracle8i. Le niveau actuel est 8.1.5.0.2 : il s'applique aux éditions Enterprise et Standard d'Oracle8i. **$ORACLE_HOME/bin** doit se trouver dans votre chemin pour que le correctif

© copyright Éditions Eyrolles

soit pris en compte. Voici quelques conseils à propos des fichiers correctifs, en cas de problème. Les versions antérieures des correctifs d'Oracle8 se décompressaient directement dans le répertoire de travail courant, ce qui n'est pas très propre. La plupart des archives **tar** créent un sous-répertoire et y décompressent les fichiers. Avec Oracle sous Linux, il fallait créer un sous-répertoire et transférer le correctif dans le répertoire avant de le décompresser. Le correctif 8.1.5.0.1 était même pire : il décompressait dans le répertoire de travail courant et, mystérieusement, le correctif échouait parce qu'il comptait le nombre de fichiers dans le répertoire, et qu'il trouvait un fichier en trop — le fichier de mise à jour lui-même. Il fallait alors éliminer le fichier de mise à jour compressé du répertoire avant d'installer le correctif. Oracle a fait des merveilles avec le correctif pour 8.1.5.0.2 : le correctif se décompressera dans son propre sous-répertoire ! La raison pour laquelle j'ai décrit ces problèmes est qu'il n'est pas sûr qu'ils ne ressurgissent pas dans les correctifs ultérieurs : aussi soyez attentif, et lisez toute la documentation fournie avec le correctif. De plus, il faut aussi toujours arrêter le serveur de base de données quand on installe des correctifs qui s'appliquent à lui.

1. Décompresser le correctif :

```
$ cd /home/oracle/patches
$ tar zxvf linux_815patches.tgz
```

2. S'assurer que la variable d'environnement **$ORACLE_HOME** est initialisée :

```
$ echo $ORACLE_HOME
/u01/app/oracle/product/8.1.5
```

3. Aller dans le répertoire du correctif précédemment créé :

```
$ cd linux_815patches
```

4. Lire le fichier README :

```
$ more README
```

5. Exécuter le script de shell, qui est maintenant dans le répertoire courant :

```
./linux_815patches.sh
```

Construire les scripts de création de base de données

L'assistant de configuration de base de données (**dbassist**) est utilisé pour supprimer, créer ou modifier une base de données. Dans notre cas, il y a deux raisons d'écrire des scripts de création de base de données plutôt que d'utiliser l'assistant de configuration de base de données Oracle pour configurer une

© copyright Éditions Eyrolles

base de données « de base ». La première est que l'on va probablement rencontrer des bogues, pendant le processus d'installation. La seconde, que c'est une bonne occasion de voir de quels composants est constituée la base de données. Utilisez largement l'aide, dans chacune de ces étapes, pour créer les scripts de base de données.

En exécutant **dbassist**, on voit apparaître le message : « JNLS Exception: oracle.ntp.jnls.JNLSException Unable to find any National Character Sets. Please check installation ». C'est un bogue connu : ignorer l'avertissement et cliquer sur **OK**.

1. Exécuter **dbassist** (figure 3.5).

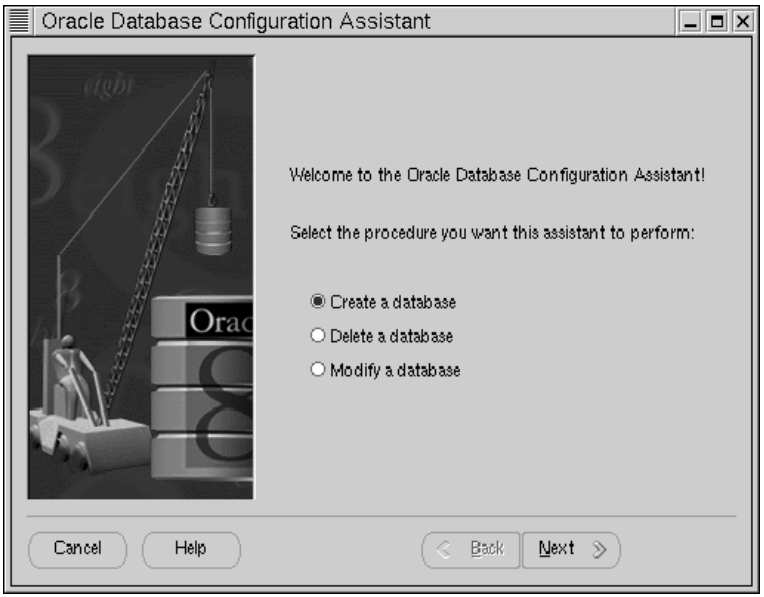

Figure 3–5
Assistant de configuration de base de données Oracle

2. Choisir **Create a database** pour créer une base de données.
3. Choisir **Custom** comme type de la base de données à créer. Typical est réputé ne pas fonctionner.
4. Choisir **Hybrid** pour le type d'environnement dans lequel la base de données va fonctionner. Si la base de données doit servir surtout à de courtes transactions de gestion, choisir le système transactionnel (OLTP, *OnLine Transaction Processing*). Si le système est destiné à établir de longs rapports, choisir le système d'aide à la décision (DSS, *Decision Support System*). En cas de doute, **Hybrid** est le meilleur choix.

© copyright Éditions Eyrolles

5. Il faut ensuite indiquer le nombre approximatif d'utilisateurs qui seront connectés simultanément à la base de données : laisser la valeur par défaut.
6. Choisir **Dedicate Mode Server** comme mode de fonctionnement par défaut de la base de données.
7. Choisir, parmi les options, celles que vous voulez configurer (voir tableau 3.3) : *Oracle interMedia*, *Oracle Visual Information Retrieval*, *Oracle interMedia text*, *Oracle Times Series*, *Oracle Spatial*, *Oracle JServer*, *Advanced Replication*, *Enable interMedia demos* et les tables d'aide de SQL*Plus. Si vous voulez simplement tester la base de données, il n'est pas nécessaire de configurer les options, qui occuperont de l'espace disque supplémentaire ainsi que de la mémoire, en particulier JServer, sachant que l'on peut les configurer plus tard. Un conseil : n'installer aucune de ces options à ce stade. Vous aurez bien assez d'occupations avec la base de données elle-même, pendant l'apprentissage.

Tableau 3–3
Options Oracle8i

Options	Description et caractéristiques
Oracle interMedia	Oracle a combiné différents produits en un seul grand produit appelé interMedia. Oracle interMedia donne à Oracle8i la possibilité de gérer des textes, des documents, des images, des documents audio, vidéo, et des informations géographiques, de manière intégrée. (Cette option n'est pas entièrement disponible sous 8.1.5).
Oracle Visual Information Retrieval	Oracle8i Visual Information Retrieval (VIR) facilite le repérage de similitudes d'images, sur la base de critères tels que la couleur, la forme et la texture.
Oracle interMedia text	Oracle interMedia text propose des outils de recherche de texte. Il indexe les documents et permet de retrouver des documents archivés. Toutes les options de recherche sont disponibles depuis n'importe quel outil PL/SQL ou SQL. interMedia est souvent appelé context Oracle.
Oracle Times Series	Oracle Times Series permet de ranger des données temporelles dans la base de données. Il dispose d'un ensemble de fonctions pour retrouver et traiter ces données. Les fonctions comprennent les calendriers, les séries temporelles et la synchronisation des heures.

© copyright Éditions Eyrolles

Tableau 3–3
Options Oracle8i

Options	Description et caractéristiques
Oracle Spatial	Les données spatiales sont des données qui se réfèrent à un localisant. Les bases de données contenant des champs géographiques, tels que les adresses, les numéros de téléphone et les codes postaux peuvent être analysées *via* un service nommé géocodification.
Oracle JServer	JServer est une machine virtuelle Java compatible avec Java2 de Oracle. On peut ranger le code Java dans la base de données et l'exécuter.
Advanced Replication	La réplication est le processus de copie et de maintenance d'objets de base de données dans les bases de données multiples, qui composent un système de base de données distribué.

8. La boîte de dialogue suivante (figure 3.6) attend un nom de base de données global et un SID. Le nom de base de données global par défaut est **dbname**, et le SID par défaut a la valeur de la variable d'environnement ORACLE_SID qui aura été initialisée précédemment : il peut s'agir de **oracl**. Le nom de base de données global concerne la base de données, alors que le SID se rapporte à l'instance qui dessert la base de données. Le nom de base de données global peut avoir deux composants : un nom de base de données et un nom de domaine ; cela ressemble aux noms d'hôte Internet. Par exemple, dans l'adresse www.oracle.com, www est le hostname, et oracle.com le domaine. Le but est de donner un nom unique, qualifié, à toute base de données Oracle dans le monde. Oracle recommande d'utiliser le domaine courant : par exemple, chez Oracle, on peut rencontrer des noms de base de données globaux tels que **db.oracle.com**. Le nom de base de données est limité à huit caractères et le domaine à 128 caractères ; la longueur du SID dépend du système d'exploitation, mais seuls huit caractères sont conservés, bien qu'il soit recommandé de n'en utiliser que quatre. En général, la partie nom de base de données du nom de base de données global est aussi le SID. Le nom de domaine n'est pas obligatoire, de sorte que l'on peut simplement entrer le même nom dans les deux champs, par exemple **orcl** et **orcl**. Si les champs apparaissent vides, cliquez sur le bouton **Next** et, dans l'écran suivant, cliquez sur le bouton **Back**.

9. Vérifier le paramètre du fichier de contrôle. Il faut un fichier de contrôle pour monter et accéder à une base de données. Pour une simple base de données de test, utiliser les paramètres par défaut. Si vous initialisez votre

© copyright Éditions Eyrolles

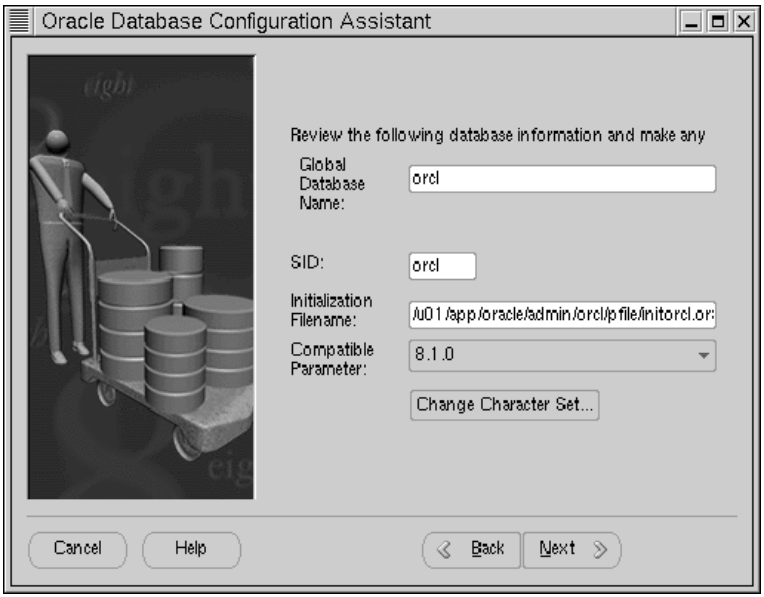

Figure 3–6
Paramètres d'initialisation de l'assistant de configuration de base de données

système de fichiers avec le système OFA de Oracle, vous avez la possibilité de changer le chemin du répertoire des fichiers de contrôle en **/u02/oradata/orcl/control01.ctl** et **/u03/oradata/orcl/control01.ctl**.

10. L'écran suivant contient un tableau qui sert à renseigner SYSTEM, USER, ROLLBACK, INDEX et TEMPORARY. Ce sont des tablespaces à usage spécifique. Le tablespace SYSTEM contient la plupart des tables de l'administrateur et le dictionnaire des données, USER les objets utilisateur, ROLLBACK les segments de rollback, INDEX les index, et le tablespace TEMPORARY, les tables provisoires. Remarquer la taille initiale de chaque tablespace et son emplacement ; les options par défaut sont satisfaisantes pour une base de données de test et, si vous avez configuré votre système de fichiers avec OFA, le tableau 3.4 donne les emplacements des répertoires.

Tableau 3–4
Emplacements des fichiers de données des tablespaces

Tablespace	Emplacement
SYSTEM	/u02/oradata/orcl/system01.dbf
USER	/u02/oradata/orcl/user01.dbf

© copyright Éditions Eyrolles

Tableau 3–4
Emplacements des fichiers de données des tablespaces

Tablespace	Emplacement
ROLLBACK	/u02/oradata/orcl/rbs01.dbf
INDEX	/u02/oradata/orcl/indx01.dbf
TEMPORARY	/u02/oradata/orcl/temp01.dbf

11. L'écran suivant contient des informations sur les journaux (*logs redo*). Augmenter la taille des journaux à 5 000 Ko (5 Mo). Les journaux sont obligatoires pour ouvrir la base de données. Chaque transaction est enregistrée dans un journal — utilisé pour la restauration des données. Si vous avez configuré votre système avec OFA, ranger les journaux dans les emplacements suivants :
/u03/oradata/orcl/redo01.log et **/u03/oradata/orcl/redo02.log**.
12. Vérifier le paramètre de connexion. Les journaux sont commutés dans certaines circonstances. L'intervalle entre les vérifications (checkpoint) garantit qu'ils seront commutés au moins une fois.
13. L'écran suivant affiche des informations globales du système relatives à l'usage de la mémoire. Le SGA influence beaucoup les performances de la base de données. Une base de données de production exigera un suivi et des ajustements du SGA. Les paramètres par défaut sont suffisants pour une base de données d'essai.

Important. La taille du bloc de base de données doit être spécifiée à l'installation de la base de données, et ne peut plus être changée par la suite.

14. Vérifier le chemin de débogage, dans lequel sont écrits les fichiers de trace pour l'utilisateur et les processus en arrière-plan.
15. Sauvegarder les informations dans un script de shell au lieu de laisser l'assistant de base de données Oracle créer la base de données. Choisir **Done** quand c'est fait. Sauvegarder le script dans le chemin par défaut, c'est-à-dire **/u01/app/oracle/product/8.1.5/install/sqlorcl.sh**.

Exécuter les scripts de création de base de données

Examiner le fichier **$ORACLE_HOME/install/sql<SID.>sh**, qui est utilisé pour appeler d'autres fichiers, individuellement. Exécuter le fichier pour commencer la construction de la base de données.

© copyright Éditions Eyrolles

```
$ cd $ORACLE_HOME/install
$. /sqlorcl.sh
```

C'est le moment de prendre un café ou un thé : il faut attendre patiemment la nouvelle base de données. La construction d'une instance complète, avec toutes les options, demande de vingt minutes à plus d'une heure.

Post-installation

Maintenant qu'Oracle est installé, vous devez laisser la situation se décanter un peu avant de commencer à vous en servir. Avant cela, vous devez peaufiner certains points, par exemple changer certains mots de passe critiques, mettre à jour un nouveau format de date, et configurer Linux pour démarrer et arrêter Oracle automatiquement.

Comptes SYS et SYSTEM

Sous Oracle, il y a trois comptes à usage particulier, créés par défaut : interne, SYS et SYSTEM. Le compte interne est un compte spécial, souvent utilisé par les scripts. Le compte SYS est propriétaire des tables de l'administrateur et du dictionnaire des données. On est rarement amené à utiliser le compte SYS. Le compte SYSTEM est propriétaire de toutes les tables de l'administrateur et des vues créées après l'installation de la base de données, comme celles qui sont utilisées par les outils Oracle. Le compte SYSTEM n'a pas de rapport direct avec le tablespace SYSTEM, même s'il a le même nom. Les mots de passe des comptes SYSTEM et SYS seront changés. Dans l'exemple suivant, remplacer **newpassword** par un mot de passe de votre choix.

Pour changer les mots de passe, se connecter en se servant de **svrmgrl** :

```
[oracle@dbsvr oracle]$ svrmgrl

Oracle Server Manager Release 3.0.5.0.0 - Production
(c) Copyright 1997, Oracle Corporation.  All Rights Reserved.
Oracle8 Release 8.0.5.1.0 - Production
PL/SQL Release 8.0.5.1.0 - Production

SVRMGR> connect internal
Connected.
SVRMGR> alter user sys identified by newpassword ;
Statement processed.
SVRMGR> alter user system identified by newpassword ;
Statement processed.
SVRMGR>
```

© copyright Éditions Eyrolles

Toutes les tables de base, ainsi que les vues du dictionnaire des données, sont rangées dans le schéma SYS. Ne modifier aucun objet du schéma SYS ; cela ne doit être fait que par Oracle. Le compte SYSTEM reçoit tous les privilèges système pour la base de données, et sert à créer les tables additionnelles ou les vues nécessaires à l'administration de la base de données. Le compte SYSTEM est généralement propriétaire des tables et des vues utilisées par les outils Oracle.

Configuration du fichier de paramètres

Au démarrage, Oracle se sert d'un fichier appelé *fichier des paramètres d'initialisation*. Le fichier s'appelle **initSID.ora** et on le désigne couramment par **INIT.ORA**. Remplacez le SID par votre **$ORACLE_SID**. Dans Oracle8 le fichier est situé dans **/u01/admin/$ORACLE_SID/pfile**. Dans Oracle8i, il se trouve dans **/u01/app/oracle/admin/$ORACLE_SID/pfile**.

Le fichier d'initialisation peut contenir des paramètres qui optimisent les performances de la base de données, des options par défaut ou des limites, et spécifier des noms de fichiers. Les paramètres qui définissent les structures de mémoire SGA sont dans le fichier **INIT.ORA.** Le fichier **INIT.ORA** par défaut, installé avec une nouvelle base de données, comprend trois groupes d'options étiquetés SMALL, MEDIUM et LARGE. En général, les options MEDIUM et LARGE sont placées en commentaires. Les paramètres se calculent d'après la valeur de la RAM installée et peuvent être augmentés. Il n'entre pas dans le cadre de ce chapitre de décrire la grande variété des options d'initialisation. Un conseil : démarrer avec les options par défaut, à part le paramètre de format de la date ! La sortie standard pour les dates dans le serveur Oracle contient des années à deux chiffres ; par exemple, 10-AVR-99. Pour avoir des années de quatre chiffres, ajouter le paramètre **nls_date_format** dans le fichier **INIT.ORA**. Editer le fichier **INIT.ORA** et ajouter la ligne suivante :

```
nls_date_format = "YYYY-MM-DD"
```

Démarrage et arrêt automatiques

Il est pratique de démarrer et d'arrêter Oracle automatiquement, en même temps que Linux. Oracle contient un script appelé **dbstart** utilisé dans les scripts de contrôle de l'exécution. **dbstart** consulte le fichier **/etc/oratab** pour connaître les instances démarrées et arrêtées automatiquement.

```
# more /etc/oratab
# The last parameter tells dbstart whether it should start and stop Oracle (Y/N)
orcl:/u01/app/oracle/product/8.0.5:Y
```

© copyright Éditions Eyrolles

Un niveau d'exécution (*run-level*) est un état logique qui définit une configuration. Un système peut fonctionner différemment selon qu'il est de niveau 3 ou de niveau 4. Un système Red Hat fonctionne normalement aux niveaux 3 ou 5. Le niveau 3 est considéré comme le mode multi-utilisateur complet, alors que le niveau 5 ne fournit qu'un écran de login X Window. Sous Red Hat Linux, les scripts de contrôle d'exécution sont rangés sous **/etc/rc.d/init.d**. Des liens symboliques sont créés sous ..**/rc0.d**, **../rc1.d**, **../rc2.d**, **../rc3.d**, **../rc4.d**, **../rc5.d**, et ..**/rc6.d**, selon le niveau où se situe l'action. Les liens symboliques commencent par un S ou un K, selon qu'ils démarrent (*start*) ou qu'ils interrompent (*kill*), et sont suivis d'un numéro qui indique l'ordre d'exécution des scripts dans le répertoire. La commande **telinit** sert à changer le niveau d'exécution de Linux. Voir **man init** pour plus d'informations.

Vous pouvez utiliser le script suivant comme point de départ. Copiez le contenu dans un fichier appelé **oracle**, dans le répertoire **/etc/rc.d/init.d.** Vous devez aussi le rendre exécutable avec la commande : **oracle u+x chmod**. Vérifiez aussi que la variable d'environnement **ORA_HOME** est conforme à votre configuration.

```
# !/bin/bash
# Start and stop the Oracle Instance
#
# chkconfig: 345 91 19
# description: Starts the Oracle listener and instance

ORA_HOME=”/u01/app/oracle/product/8.1.5”
ORA_OWNER=”oracle”

if [ ! -f $ORA_HOME/bin/dbstart -o ! -d $ORA_HOME ]
then
        echo "Oracle startup: cannot start"
        exit 1
fi

 case "$1" in
  start)
     # Startup the Oracle listener and instance
     echo -n "Oracle startup: "
     su - $ORA_OWNER -c "$ORA_HOME/bin/lsnrctl start"
     su - $ORA_OWNER -c $ORA_HOME/bin/dbstart
     touch /var/lock/subsys/oracle
     echo "Finished"
     ;;
  stop)
# Shutdown the Oracle listener and instance
     echo -n "Oracle shutdown: "
     su - $ORA_OWNER -c "$ORA_HOME/bin/lsnrctl stop"
     su - $ORA_OWNER -c $ORA_HOME/bin/dbshut
```

© copyright Éditions Eyrolles

```
    rm -f /var/lock/subsys/oracle
    echo "Finished"
    ;;
 reload|restart)
   $0 stop
   $0 start
    ;;
 *)
   echo "Usage: /etc/rc.d/init.d/oracle {start|stop|restart|reload}"
   exit 1
esac
exit 0
```

La ligne # **chkconfig : 345 91 19** est destinée au programme **chkconfig.** Le nombre **345** indique à quel niveau le script sera exécuté, le nombre **91** dans quel ordre démarre Oracle, et **19** dans quel ordre l'arrêter. Le programme **chkconfig** créera des liens symboliques du script Oracle vers les divers répertoires des niveaux d'exécution. Voir **man chkconfig** pour plus d'informations à propos de l'utilitaire **chkconfig** de Red Hat.

```
#chkconfig –add oracle
```

Configuration du réseau

Le réseau sous Oracle8/8i s'appelle Net8. Oracle8 et Oracle8i utilisent des fichiers de configuration réseau similaires. Il n'y a aucun utilitaire d'assistance à la configuration du réseau dans Oracle8 : on les modifie à la main. Oracle8i met à disposition un assistant écrit en Java, appelé **Network Assistant**, qui configure le réseau.

Le serveur Oracle propose beaucoup de protocoles réseau. Les deux méthodes les plus courantes de connexion aux serveurs Oracle en réseau consistent à donner des noms d'hôte TCP/IP ou des noms de service TNS (*Transport Network Substrate*). Si l'on a choisi d'installer une seule instance sur le serveur, adopter les noms d'hôte TCP pour se connecter à distance : par exemple, si le serveur Linux a le nom d'hôte **fred**, utiliser **fred** comme chaîne de connexion, dans le nom de service, pour se connecter à l'instance. Un serveur qui exécute des instances multiples d'Oracle ne peut pas utiliser les noms d'hôte TCP/IP. TNS fait correspondre des noms de service aux hôtes qui exécutent des protocoles différents, dont TCP/IP : dans ce cas, le nom d'instance peut être spécifié. Les noms TNS sont rangés dans le répertoire **$ORACLE_HOME/network/admin**, dans le fichier **tnsnames.ora**.

© copyright Éditions Eyrolles

Net8 Assistant – /u01/app/oracle/product/8.1.5/network/admin/
File Edit Command Help
Network
Profile
Net Service Names
Listeners
Oracle Names Servers
Net Service Names are used as a simple way to identify a service, such as a database, on the network.
A Net Service Name includes settings which specify the service name, location, and any special options necessary to locate a service over the network. An end user enters a connect string that uses a Net Service Name rather than a more lengthy description of the service:
connect username/password@net_service_name
To see if net service names have been created on this host: Double-click the Net Service Names folder. If no Net Service Names exist, click "+" on the toolbar or Create from the Edit menu.
NOTE: The Net8 Assistant does not support some older service naming syntax. These older entries will continue to function but may not be displayed by the Net8 Assistant.

Figure 3–7
Assistant Net8

Voici un exemple d'entrée de **tnsnames.ora** qui spécifie l'hôte 192.168.10.1, fonctionnant sur le port 1521, avec le nom d'instance **orcl**.

```
orcl =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL= TCP)(Host= 192.168.10.1)(Port= 1521))
    (CONNECT_DATA = (SID = orcl))
  )
```

Oracle8i propose l'assistant Net8, pour configurer le réseau (voir figure 3.7). Si vous n'avez qu'un seul serveur, continuez à utiliser les noms d'hôte TCP pour accéder à votre serveur. Certaines sociétés utilisent les *Oracle Names Servers*, des référentiels centralisés de noms de service. Les clients peuvent utiliser le *Oracle Names Server* comme ressource pour rechercher des instances Oracle. Deux applications Java — **netasst** et **netca** — servent à configurer le réseau sous Oracle8i.

© copyright Éditions Eyrolles

Utiliser Oracle8/8i

Oracle est installé, vous êtes prêt à vous en servir. Cette section décrit comment démarrer et arrêter le serveur, quels processus sont nécessaires au fonctionnement de la base de données, explique comment créer un compte ou utiliser SQL*Plus, donne une vue d'ensemble du dictionnaire des données ainsi qu'une brève introduction sur la manière de charger des données dans la base de données, ou d'en extraire.

Démarrage et arrêt

Oracle Server Manager, **svrmgrl**, et SQL*Plus peuvent démarrer et arrêter l'instance. La fonctionnalité du Gestionnaire de serveur étant intégrée dans SQL*Plus, Oracle envisage de ne plus distribuer l'utilitaire à l'avenir. Je préfère **svrmgrl** à SQL*Plus pour exécuter des tâches d'administration de base de données de base ; par conséquent, cette section décrira brièvement cet outil. Le **l** dans **svrmgrl** signifie en mode ligne. Quelques plates-formes disposent d'une version graphique du Gestionnaire de serveur ; Linux n'en fait pas partie ; SQL*Plus sera décrit plus loin dans ce chapitre. Le Gestionnaire de serveur comprend un groupe de commandes d'administration de la base de données. Les commandes **STARTUP**, **SHUTDOWN** et **SGA SHOW** sont présentées dans le tableau 3.5.

Tableau 3–5
Commandes STARTUP et SHUTDOWN du Gestionnaire de serveur

Commande	Description
STARTUP	Démarrer l'instance, monter la base de données et ouvrir la base de données.
STARTUP NOMOUNT	Démarrer l'instance. Rarement utilisée.
STARTUP MOUNT	Démarrer l'instance et monter la base de données. La base de données n'est pas ouverte. Utilisée pour certaines tâches d'administration.
STARTUP OPEN RECOVER	Démarrer l'instance, monter la base de données, ouvrir la base de données, et commencer la restauration. On a, en retour, un message d'erreur si la restauration n'est pas nécessaire. D'habitude, la restauration est automatique.
SHUTDOWN NORMAL	Oracle attend que toutes les connexions soient fermées pour s'arrêter
SHUTDOWN TRANSACTIONAL	Oracle attend que toutes les transactions se terminent et ferment toutes les connexions.

© copyright Éditions Eyrolles

Tableau 3–5
Commandes STARTUP et SHUTDOWN du Gestionnaire de serveur

Commande	Description
SHUTDOWN IMMEDIATE	Toutes les transactions actives sont annulées (*rollback*), et les connexions sont fermées. Certaines instructions SQL peuvent être abandonnées.
SHUTDOWN ABORT	Les instructions SQL en cours sont terminées, les transactions non validées ne sont pas annulées (*rollback*), et toutes les connexions sont fermées.

Une instance peut être démarrée par la commande **STARTUP** et arrêtée par la commande **SHUTDOWN.** L'instance est démarrée sans que la base de données ne soit réellement montée à l'aide de la commande **STARTUP NOMOUNT**, qui ne sert normalement que pendant la création de la base de données. La base de données peut aussi être démarrée et montée, mais sans être ouverte, par la commande **STARTUP MOUNT**. Ce procédé est utile si l'on veut renommer des fichiers, activer ou désactiver des options d'archivage dans les journaux ou exécuter une restauration complète de la base de données. Pour ouvrir une base de données, utiliser la commande **ALTER DATABASE OPEN**. La commande **STARTUP RESTRICT** sert à ouvrir la base de données en mode restreint, qui réserve l'accès à la base de données aux utilisateurs qui disposent des privilèges **CREATE SESSION** et **RESTRICTED SESSION**. On sélectionne un fichier de paramètre à l'aide de la commande **STARTUP PFILE=initSID.ora.** Dans la plupart des cas, la commande **STARTUP** de base sera suffisante pour démarrer la base de données.

```
SVRMGR> help
The following are SIMPLIFIED syntax descriptions. For complete syntax
descriptions, please refer to the Oracle Server Manager User's Guide.

STARTUP      [DBA] [FORCE] [PFILE=filespec] [EXCLUSIVE | SHARED]
          [MOUNT dbname | OPEN dbname] [NOMOUNT]

SHUTDOWN     [NORMAL | IMMEDIATE | ABORT]

MONITOR For graphical modes only, bring up a monitor
ARCHIVE LOG  [START] [STOP] [LIST] [NEXT] [<n>] [ALL] ['destination']

RECOVER      { [DATABASE [MANUAL] ] | [TABLESPACE ts-name, [tsname]] }

CONNECT      [username [/password] ] [INTERNAL] ['@'instance-spec]
DISCONNECT
```

© copyright Éditions Eyrolles

```
SET options: INSTANCE, ECHO, TERMOUT, TIMING, NUMWIDTH, CHARWIDTH
SHOW LONGWIDTH, DATEWIDTH, AUTOPRINT and for SHOW: ALL, SPOOL

EXIT
REM
        SQL statements can also be executed.
```

L'instance peut être arrêtée par la commande **SHUTDOWN.** Le problème le plus délicat à traiter au moment de l'arrêt est celui des transactions ouvertes. Si les utilisateurs sont connectés à l'instance pour des transactions ouvertes, il faut savoir comment traiter les transactions. La commande **SHUTDOWN NORMAL** attend que tous les utilisateurs en cours soient déconnectés de la base de données et, dans ce cas, toutes les transactions se terminent normalement. La commande **SHUTDOWN IMMEDIATE** annule (*rollback*) toutes les transactions non validées (*commit*) et déconnecte toutes les connexions à l'instance. Comme son nom ne l'indique pas, cela peut prendre du temps pour annuler toutes les transactions non validées. Certains processus sont autorisés à se terminer avant déconnexion. La commande **SHUTDOWN TRANSACTIONAL** autorise les transactions actives à se terminer ; elle épargne aux utilisateurs une perte de travail, et les déconnecte de l'instance une fois que toutes les transactions sont terminées. Cela peut prendre du temps si les utilisateurs laissent ouvertes des transactions non validées. Enfin, la commande de dernier recours est **SHUTDOWN ABORT**. Cette commande ferme l'instance instantanément : les instructions SQL des clients sont abandonnées, les transactions sont interrompues (sans rollback) et tous les utilisateurs sont déconnectés. Cette commande exigera une restauration de la base de données, au prochain démarrage, ce qui se produira automatiquement. Si ce n'est pas le cas, utiliser la commande **STARTUP OPEN RECOVER.** Une fois que l'une des commandes **SHUTDOWN** a été soumise, aucune nouvelle connexion n'est autorisée. Pour plus d'informations sur le Gestionnaire de serveur, taper **HELP** à l'invite du Gestionnaire de serveur.

Pour mieux connaître l'usage de la mémoire dite zone globale système, utilisez la commande **SHOW SGA** de SQL*Plus ou du Gestionnaire de serveur, qui donne la taille totale de la zone globale système, la taille fixe, la taille variable, les buffers de base de données et les buffers des journaux.

```
SVRMGR> connect / as sysdba
Connected.
SVRMGR> show sga
Total System Global Area 8154384 bytes
Fixed Size 48400 bytes
Variable Size 6905856 bytes
Database Buffers 1126400 bytes
Redo Buffers 73728 bytes
```

© copyright Éditions Eyrolles

Processus en arrière-plan

Quand Oracle fonctionne, on voit tout de suite que beaucoup de processus différents se déroulent. Faisons rapidement le tour des fonctions par processus. Passer une commande **ps**, qui montre bien que le nom de chaque processus contient le SID de l'instance.

```
 [oracle@dbsvr oracle]$ ps aux|grep ora_
oracle 3798 0.0 4.4 114916 5656 ?      S 12:38 0:00 ora_pmon_orcl
oracle 3800 0.0 4.5 114884 5828 ?      S 12:38 0:00 ora_dbw0_orcl
oracle 3802 0.0 4.2 114368 5440 ?      S 12:38 0:00 ora_lgwr_orcl
oracle 3804 0.0 4.0 114348 5152 ?      S 12:38 0:00 ora_ckpt_orcl
oracle 3806 0.0 9.6 114388 12316 ?     S 12:38 0:02 ora_smon_orcl
oracle 3808 0.0 5.4 114396 7024 ?      S 12:38 0:00 ora_reco_orcl
oracle 3810 0.0 3.6 114300 4732 ?      S 12:38 0:00 ora_s000_orcl
oracle 3812 0.0 3.9 114416 5008 ?      S 12:38 0:00 ora_d000_orcl
oracle 3814 0.0 4.0 114468 5136  ?     S 12:38 0:00 ora_arc0_orcl
```

Il se peut que l'on n'ait pas le même ensemble de processus se déroulant sur une autre machine. Il y a au moins cinq processus Oracle dont on est sûr : l'écriture de la base de données, l'écriture du journal, le moniteur système, le moniteur de processus et le processus de checkpoint. Remarquer que le nom de chaque processus est précédé par **ora**, et finit par le nom d'instance.

L'écriture de la base de données (DBW0) est responsable de l'écriture des données modifiées dans les fichiers de données physiques. Quand les données changent, elles sont placées dans le cache de la base de données. À intervalles donnés, l'écriture écrit les buffers modifiés depuis le cache sur les fichiers de la base de données. L'écriture de la base de données doit s'assurer qu'il y a suffisamment de buffers disponibles dans le cache. Les performances de la base sont améliorées du fait que les modifications se font dans un buffer mémoire au lieu de se faire sur le disque. Des processus d'écritures de base de données multiples peuvent améliorer les performances, bien qu'en général, un seul suffise.

Le processus d'écriture du journal (LGWR) enregistre les modifications, inscrites dans le cache de journalisation, dans les fichiers journaux. L'écriture du journal exécute des écritures séquentielles dans le fichier journal dans les cas suivants : quand le cache du journal est plein au tiers, quand une alarme se produit (toutes les trois secondes), quand une transaction est validée et avant que l'écriture de la base de données n'écrive les blocs, modifiés dans le cache de la base, dans les fichiers de données.

Le moniteur système (SMON) vérifie la cohérence de la base de données à l'ouverture et commence la restauration, si nécessaire. Le moniteur de processus (PMON) récupère les ressources des processus en échec. Le processus de checkpoint (CKPT) est chargé de mettre à jour les informations d'état de la

© copyright Éditions Eyrolles

base de données, à chaque fois que des changements dans le cache sont enregistrés.

D'autres processus, optionnels, prennent en charge l'archivage, la récupération, la régulation et le serveur partagé. Le processus d'archivage (ARCn) copie les journaux en ligne vers la mémoire de stockage, quand ils sont pleins ou quand il y a commutation. L'archivage des journaux s'inscrit dans la stratégie de protection des bases de données. S'il se produit un *crash* de la base et que les fichiers de données sont perdus, on repart de la sauvegarde pour restaurer la base de données telle qu'elle était à un certain moment et on utilise conjointement les journaux archivés pour rétablir l'état de la base de données à un moment ultérieur. Si la base de données est en mode journalisation et que le processus d'archivage ne peut trouver d'espace libre, toutes les transactions de la base de données seront abandonnées.

Le processus de récupération (RECO) s'utilise dans une configuration de base de données distribuée pour résoudre les transactions en attente, phénomène dû à un échec du réseau ou du système. Le régulateur (Dnnn) est utilisé dans les configurations de serveurs multithread. Il achemine les requêtes de chaque processus utilisateur vers les processus du serveur partagé et renvoie les réponses au processus utilisateur. Il y a au moins un régulateur pour chaque protocole de communication. Le processus de serveur partagé (Snnn) sert dans une configuration de serveur multithread, pour répondre aux requêtes des clients.

Il y a encore trois processus que vous ne rencontrerez probablement pas : le verrouillage (LCK0), la file des travaux (SNPn) et le moniteur de queue (QMNn). Ces processus sont utilisés respectivement par *Oracle Parallel Server*, le rafraîchissement instantané des tables et *Advanced Oracle Queuing*.

Chaque processus gère son propre ensemble de tâches. Il arrive qu'un processus entre en conflit avec un autre, pour une ressource. Par exemple, les écritures du journal et de la base de données qui essaieraient d'accéder à un lecteur en même temps. De telles situations entrent en ligne de compte dans un environnement de production conséquent.

Créer un compte

Il y a trois choses à savoir faire quand on crée un compte utilisateur. La première est de créer le compte lui-même, la seconde de lui donner des droits et la troisième, de lui allouer un quota sur un ou plusieurs tablespaces. Il y a beaucoup de choses à connaître pour traiter la sécurité dans Oracle. Les instructions suivantes vous aideront à créer un compte de connexion de base sans trembler. Il faut être connecté avec le privilège DBA pour créer des

© copyright Éditions Eyrolles

comptes utilisateur, et d'abord utiliser le compte SYSTEM pour créer son propre compte. La commande suivante crée un utilisateur nommé **pat** avec le mot de passe **er34533**. L'utilisateur a accès au tablespace par défaut **users** et au tablespace temporaire **temp**. Il est important de spécifier un tablespace par défaut car il ne faut pas que les utilisateurs créent des objets dans le tablespace SYSTEM. La dernière ligne fournit un espace utilisateur non limité sur le tablespace **users**. Il serait préférable de limiter le quota en spécifiant une taille telle que **50M**, pour 50 Mo, ou **500 K** pour 500 Ko.

```
SQL> CREATE USER pat
  2 IDENTIFIED BY er34533
  3 DEFAULT TABLESPACE users
  4 TEMPORARY TABLESPACE temp
  5 QUOTA unlimited ON users ;
```

Une fois le compte créé, il faut lui donner les droits pour la connexion à la base de données. Il y a deux types de privilèges : système et objet. Les privilèges système donnent aux utilisateurs la possibilité d'exécuter dans la base de données des actions telles que la création, la suppression et la mise à jour des tables, des vues, des segments de rollback et des procédures. Les privilèges d'objet donnent aux utilisateurs la possibilité d'accéder et de modifier des objets de la base de données tels que les tables, les vues, les séquences, les procédures, les fonctions ou les paquetages. Il y a 80 privilèges système. La commande **GRANT** accorde un privilège à un utilisateur et la commande **REVOKE** a l'effet contraire. La vue **DBA_SYS_PRIVS** liste les privilèges système accordés aux utilisateurs. Un utilisateur peut identifier les privilèges système disponibles dans la session courante en consultant la vue **SESSION_PRIVS.** Comme il y a beaucoup de privilèges individuels, il s'avère difficile d'entretenir des privilèges multiples sur un grand nombre de comptes utilisateur. Dans cette optique, Oracle gère les privilèges en introduisant la notion de *rôles*. Les privilèges peuvent être assignés aux rôles, qui à leur tour sont accordés aux utilisateurs. Il y a une grande variété de rôles prédéfinis dans Oracle. Les trois plus connus sont CONNECT, RESOURCE et DBA. Le rôle CONNECT donne les droits d'accès et d'utilisation de la base de données suffisants pour la plupart des utilisateurs. Le rôle RESOURCE fournit des droits complémentaires, utiles pour les programmeurs de base de données. Le rôle DBA a tous les privilèges. C'est la commande **GRANT** qui sert à assigner un rôle à un utilisateur. Un utilisateur peut être membre de plusieurs rôles.

```
SQL> GRANT CONNECT TO pat ;
```

© copyright Éditions Eyrolles

L'administrateur change le mot de passe d'un utilisateur à l'aide de la commande **ALTER USER IDENTIFIED**, et l'utilisateur se sert de la commande **PASSWORD** pour changer le mot de passe de son compte.

```
SQL> alter user pat identified by secret_pwd ;

User altered.

SQL> password
Changing password for PAT
Old password:
New password:
Retype new password :
```

On consulte les connexions actives à l'aide de la commande DBA suivante : **SELECT * from v$session**. Pour arrêter une session immédiatement, utiliser la commande **ALTER SYSTEM KILL SESSION sid,session**. On connaît le **sid** et la **session** à partir de **v$session**. Pour déconnecter une session à la fin d'une transaction, utiliser la commande **ALTER SYSTEM DISCONNECT sid, session POST_TRANSACTION**. Supposons, par exemple, qu'à l'aide d'une requête **SELECT** sur **v$session**, on a pu constater qu'il existe un enregistrement pour l'utilisateur **pat** contenant un SID de 7, et une SESSION de 15 : on exécutera la commande **ALTER SYSTEM KILL SESSION '7,15'**.

SQL*Plus

SQL*Plus est l'interface d'origine vers le serveur Oracle fournie par Oracle sur la plate-forme Linux. C'est une application de type terminal qui accepte des instructions SQL et des blocs PL/SQL. Doté de nombreuses fonctions, dont la possibilité de formater la sortie des requêtes SQL, SQL*Plus a l'air simple mais, en réalité, il est fonctionnellement très riche. Des livres entiers lui sont dédiés. Cette section décrira brièvement certaines de ses caractéristiques les plus importantes. SQL*Plus sera installé dans le répertoire **$ORACLE_HOME/bin.** Pour démarrer l'application, il suffit de taper **sqlplus**, ou d'ajouter une chaîne de connexion, **sqlplus pat/er34533.**

Important. La chaîne de connexion sert à se connecter à une base de données et dépend du protocole du réseau. Les plus courants sont les noms TNS et les noms d'hôte TCP/IP. Leur format standard est user/password@net_service_name. La plupart des applications admettront user@net_service_name et demanderont le mot de passe. Par exemple, on accède à une instance nommée **orcl** à l'aide de la chaîne de connexion scott/tiger@orcl. Si une seule instance est présente sur un serveur, on se sert tout simplement du nom d'hôte TCP/IP du serveur, en tant que nom de service.

© copyright Éditions Eyrolles

Remarque : dans SQL*Plus, l'invite par défaut est **SQL>**. Vous pouvez en changer à l'aide de la commande **SET : SET SQLPROMPT *monprompt***>. Pour sortir de SQL*Plus, taper simplement **EXIT**. Les commandes SQL*Plus sont indifférentes à la casse. Dès que l'on voit le prompt SQL>, entrer des instructions SQL : une instruction doit finir par un point-virgule ou une barre oblique (/) sur la ligne elle-même, ou par une ligne blanche ; la ligne blanche termine l'entrée de la commande SQL et la range dans un buffer, mais elle ne soumet pas la requête.

Toutes les commandes SQL sont rangées dans le buffer. On réexécute le buffer, en entrant une barre oblique (/) ou la commande **RUN** ; on voit son contenu avec la commande **LIST**. Pour éditer une commande, taper **LIST** : chaque ligne aura un numéro de ligne ; trouver la ligne à changer et taper son numéro dans SQL*Plus. La ligne devient active : utiliser la commande **CHANGE** pour modifier le texte. Par exemple, pour changer le mot **all_objects** en **user_objects**, entrer la commande **change/all_/user_**.

Le buffer est facile d'utilisation, mais il existe un meilleur moyen d'éditer des commandes : l'éditeur. La variable SQL*Plus **_editor** contient le nom de l'éditeur qui sera ouvert par SQL*Plus. La commande **DEFINE** permet d'initialiser la valeur de **_editor**. Pour voir la valeur de **_editor**, tapez simplement **define _editor** ; pour changer la valeur, tapez **define _editor=vi**.

```
SQL>define _editor
DEFINE _EDITOR = "ed" (CHAR)
SQL>define _editor=vi
SQL>define _editor
DEFINE _EDITOR = "vi" (CHAR)
```

Utiliser la commande **EDIT** pour placer le contenu du buffer dans l'éditeur externe. Oracle range le contenu du buffer dans un fichier **afiedt.buf**, ouvert par l'éditeur. Une fois terminées les modifications des instructions SQL, les sauvegarder, sortir de l'éditeur et revenir à SQL*Plus : le buffer reflétera alors les changements effectués dans l'éditeur ; passer la commande **RUN** pour exécuter l'instruction. Il est courant de terminer une instruction SQL ainsi éditée par une barre oblique (/), sur une ligne.

```
SELECT user_name, city, state FROM customer_info
/
```

Le contenu du buffer est sauvegardé dans un fichier par la commande **SAVE.** L'extension **.sql** sera automatiquement associée au nom du fichier. On remplace le buffer par le contenu d'un fichier avec la commande **GET.** Pour exécuter un script immédiatement, utiliser la commande @ suivie du nom de fichier, au lieu de le placer simplement dans le buffer.

© copyright Éditions Eyrolles

Le fichier doit avoir l'extension **.sql** pour être reconnu.

```
SQL> SAVE nom_de_fichier
SQL> GET nom_de_fichier
SQL> @nom_de_fichier
```

La commande **HOST** peut servir à passer des commandes à Linux. Par exemple, pour consulter un répertoire dans SQL*Plus, utiliser la commande **HOST** pour exécuter la commande **ls.** Un raccourci pour **HOST** est le point d'exclamation (**!**) : SQL> **!ls.**

```
SQL> host ls
init.sql loop.sql
SQL> !ls
init.sql loop.sql
```

On range les préférences dans un fichier, **login.sql**, du répertoire par défaut d'où SQL*Plus est lancé. Il faut exécuter la commande **COMMIT** après toute modification de données utilisant une instruction SQL. SQL*Plus valide automatiquement les transactions : soit après chaque instruction, soit après un certain nombre d'instructions, si l'on se sert de l'option **SET AUTOCOMMIT ON** où l'on indique explicitement le nombre d'instructions à exécuter entre les commits. Pour désactiver l'autocommit, utiliser **SET AUTOCOMMIT OFF.**

La commande **DESCRIBE**, ou **DESC**, sert à afficher la structure d'un objet.

```
SQL> describe all_objects ;
Name                          Null?    Type
_____________________________ ________ ____
OWNER                         NOT NULL VARCHAR2(30)
OBJECT_NAME                   NOT NULL VARCHAR2(30)
SUBOBJECT_NAME                         VARCHAR2(30)
OBJECT_ID                     NOT NULL NUMBER
DATA_OBJECT_ID                         NUMBER
OBJECT_TYPE                            VARCHAR2(15)
CREATED                       NOT NULL DATE
LAST_DDL_TIME                 NOT NULL DATE
TIMESTAMP                              VARCHAR2(19)

STATUS                                 VARCHAR2(7)
TEMPORARY                              VARCHAR2(1)
GENERATED                              VARCHAR2(1)
```

SQL*Plus sert aussi à démarrer et arrêter l'instance. L'instruction « **connect to Oracle as SYSDBA** » a une signification particulière : un utilisateur, connecté au serveur Oracle en tant que SYSDBA, a des privilèges spéciaux. Pour se connecter avec l'autorisation SYSDBA, ajouter le texte **as sysdba** à la fin de la chaîne de connexion. Par exemple, pour connecter le compte **pat**

© copyright Éditions Eyrolles

avec les privilèges SYSDBA, on utilisera la chaîne suivante : **connect pat/mypasswd as sysdba**. Bref, si l'on est connecté avec un compte membre du groupe dba, il suffit de taper **connect / as sysdba**. L'utilisateur doit être un membre du groupe OSDBA.

Dans les démonstrations, on trouve souvent un compte d'utilisateur **scott**, avec le mot de passe **tiger**, que tout le monde connaît, ou presque. Si vous créez ce compte, il vaut mieux ne pas laisser ce mot de passe. Certains utilitaires s'attendent à ce que le compte existe réellement, et échouent s'ils ne le trouvent pas. L'ensemble des tables DEPT, EMP, BONUS et SALGRADE sont, elles aussi, très connues. Ces tables peuvent être créées par le script **$ORACLE_HOME/rdbms/admin/utlsampl.sql**, et c'est un utilisateur ayant les privilèges DBA, ou SYSTEM, qui doit exécuter le script. Le répertoire **$ORACLE_HOME/rdbms/admin** contient beaucoup de scripts optionnels, dont la plupart contiennent des commentaires. Certains d'entre eux doivent être exécutés par un utilisateur particulier tel que SYS ou SYSTEM.

Dictionnaire des données

Le dictionnaire des données de Oracle est un référentiel central, en lecture seule des tables et des vues de référence. Oracle se sert du dictionnaire des données pour trouver la définition des objets, les allocations d'espace, les contraintes d'intégrité, les privilèges, les informations de vérification et autres. La configuration sous-jacente du dictionnaire des données est un ensemble de tables normalisées en lecture seule, plutôt difficiles à lire. Des vues accessibles à l'utilisateur en facilitent l'accès. Le dictionnaire des données appartient au schéma SYS et ne peut être modifié que par Oracle Corporation. Des modifications du schéma SYS risquent d'endommager la base de données ; par conséquent, il faut veiller à la sécurité de ce compte. Il y a beaucoup de synonymes publics relatifs aux vues du dictionnaire des données, dont, pour l'essentiel, trois ensembles de vues : ALL, USER et DBA. Il existe aussi un ensemble de tables des performances dynamiques dont le nom commence par **V$**. Les vues USER contiennent des éléments du schéma de l'utilisateur, les vues ALL tout ce qui est accessible à l'utilisateur, et les vues DBA, ce que tout utilisateur peut voir ; ces dernières sont conçues pour l'usage de l'administrateur de la base de données. La vue ALL_OBJECTS est particulièrement longue, car elle comprend tous les objets auxquels un utilisateur peut accéder.

```
SELECT * from ALL_OBJECTS ;
```

Un mot à propos de trois autres objets du dictionnaire des données : DUAL, CAT et DICT. DUAL est une table spéciale composée d'une seule colonne nommée DUMMY, qui a la valeur **X**. Elle sert à respecter la contrainte impo-

© copyright Éditions Eyrolles

sant à toute instruction SELECT de contenir un identifiant de table dans la clause FROM.

```
SELECT sysdate FROM dual ;
```

La vue CAT est un synonyme de USER_CATALOG : elle donne la liste des tables, des vues, des synonymes et des séquences appartenant à l'utilisateur.

```
SELECT * from CAT ;
```

DICT est un synonyme de DICTIONARY. La vue contient la description des vues et des tables du dictionnaire des données.

```
SQL> COLUMN table_name FORMAT a15
SQL> COLUMN comments FORMAT a50
SQL> SELECT * FROM dict WHERE table_name='DICTIONARY';

TABLE_NAME   COMMENTS
-----------  ------------------------------------------------------------
DICTIONARY   Description of data dictionary tables and views
```

Les vues des performances dynamiques sont mises à jour par le serveur Oracle pendant toute la durée de son activité. Ces tables virtuelles ne sont pas accessibles pour l'utilisateur courant, et ne sont utilisées que par l'administrateur de la base de données. Elles servent fréquemment au réglage des performance de la base elle-même, celles des performances dynamiques étant mises au point par V$.

```
SQL> SELECT * from v$version ;
BANNER
_____________________________________________________________________
Oracle8 Release 8.0.5.1.0 _ Production
PL/SQL Release 8.0.5.1.0 _ Production
CORE Version 4.0.5.0.0 _ Production
TNS for Linux: Version 8.0.5.0.0 _ Production
NLSRTL Version 3.3.2.0.0 _ Production
```

Import/export

Il existe trois utilitaires pour importer ou exporter de gros volumes de données dans la base de données Oracle : Export, Import et SQL*Loader. Chaque utilitaire a un grand nombre d'options et de caractéristiques. L'utilitaire d'exportation, appelé **exp**, extrait les définitions d'objet et les données d'une base de données Oracle et les range dans un fichier binaire, souvent appelé un dump. L'utilitaire d'importation, **imp**, insère les données extraites soit dans la même base de données, soit dans une autre. Les fichiers d'expor-

© copyright Éditions Eyrolles

WebDB

WebDB est un produit Oracle qui fournit une interface de type web pour la base de données Oracle. Au départ, WebDB est un kit de construction de sites web, écrit en PL/SQL. Il est inclus dans les bases de données et ne requiert qu'un navigateur web standard. WebDB propose quelques fonctions d'administration de base de données, dont la gestion de comptes et d'objets, ainsi que des outils simples de développement pilotés par des modèles qui permettent à l'utilisateur final de concevoir ses propres pages web, ses comptes rendus et ses formulaires, et propose des dossiers dans lesquels l'utilisateur peut ranger ses documents.

WebDB contient tous les composants nécessaires à la réalisation d'un site complet, dont PL/SQL et HTTP Listener. Quand HTTP Listener reçoit une demande, il la transmet à la passerelle PL/SQL qui transfère la demande au serveur Oracle et renvoie les résultats au client. Tous les fichiers web et les données sont rangées dans le serveur. WebDB fait appel à la sécurité standard d'Oracle, ce qui rend la gestion des comptes individuels plus facile.

L'interface avec WebDB s'appuie sur les protocoles standard d'Internet. Le client n'a pas besoin de bibliothèques spéciales ni d'un logiciel réseau pour se connecter au serveur WebDB. WebDB se fonde sur les compétences dont disposent déjà les DBA. WebDB convient pour un site intranet où les utilisateurs accèdent régulièrement à la base de données. Il sert au DBA, à l'utilisateur final et au développeur.

tation (*dump*) ne peuvent être lus que par l'utilitaire Import. Export n'est pas une méthode typique de sauvegarde d'une base de données, mais peut servir à créer une copie permanente d'un schéma, à le dupliquer sur d'autres serveurs, ou à l'archiver, si un utilisateur quitte la société. Le premier usage des utilitaires Export et Import est de passer des données d'un serveur Oracle à un autre. Il faut exécuter le script **CATEXP.SQL**, ou **CATALOG.SQL**, qui exécute **CATEXP.SQL**, pour configurer Oracle pour l'utilitaire Import. Le script ne doit être exécuté qu'une seule fois.

SQL*Loader sert à charger des données importées depuis d'autres bases de données. Il peut lire un fichier externe, contenant des données stockées dans des formats variés. Par exemple, SQL*Loader peut importer un fichier de largeur fixe, aussi bien qu'un enregistrement de longueur variable contenant des délimiteurs, tels que les tabulations, pour identifier les colonnes. SQL*Loader a besoin d'un fichier de contrôle, qui décrit les données et spécifie les fichiers de données en entrée. Il produit un fichier journal et un fichier de rejets, contenant les enregistrements rejetés.

© copyright Éditions Eyrolles

Logiciels tiers

De nombreuses applications tierces complètent Oracle sur la plate-forme Linux. Nous nous en tiendrons à quatre applications dans cette section : Orasoft SQLWork, Orasoft Object Manager, Orac et Perl DBI.

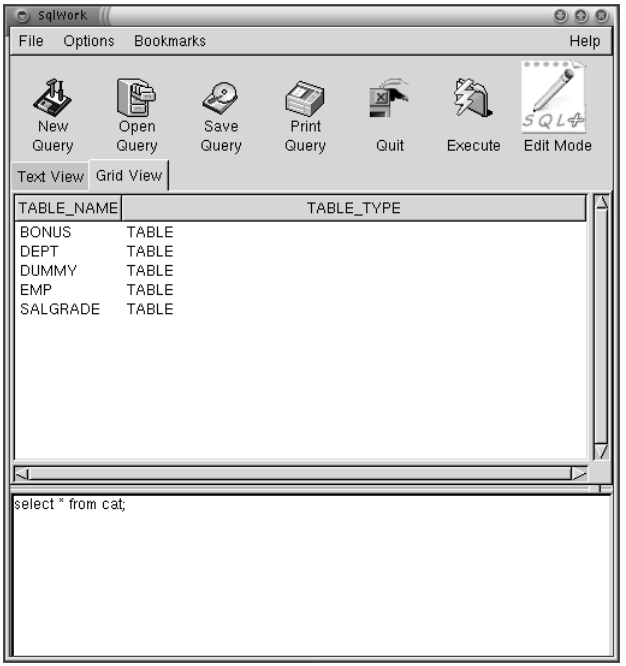

Figure 3–8
Orasoft SQLWork

Orasoft

Orasoft (www.orasoft.org) est un projet actif d'open-source qui vise à renforcer la plate-forme Linux par des applications, à la fois de grande qualité et professionnelles. A ce jour, il existe cinq applications dont SQLWork, Procedit, Session Monitor, Tables Navigator et ObjectManager. SQLWork est une feuille de travail en SQL, Procedit sert à éditer du code PL/SQL, Session Monitor à suivre les connexions aux bases de données et Tables Navigator permet de parcourir les tables Oracle. ObjectManager, le membre le plus récent de cet ensemble d'applications, est orienté vers l'administration de la base de données. Les fonctionnalités de Session Monitor et de Tables Navigator ont été reprises dans ObjectManager.

© copyright Éditions Eyrolles

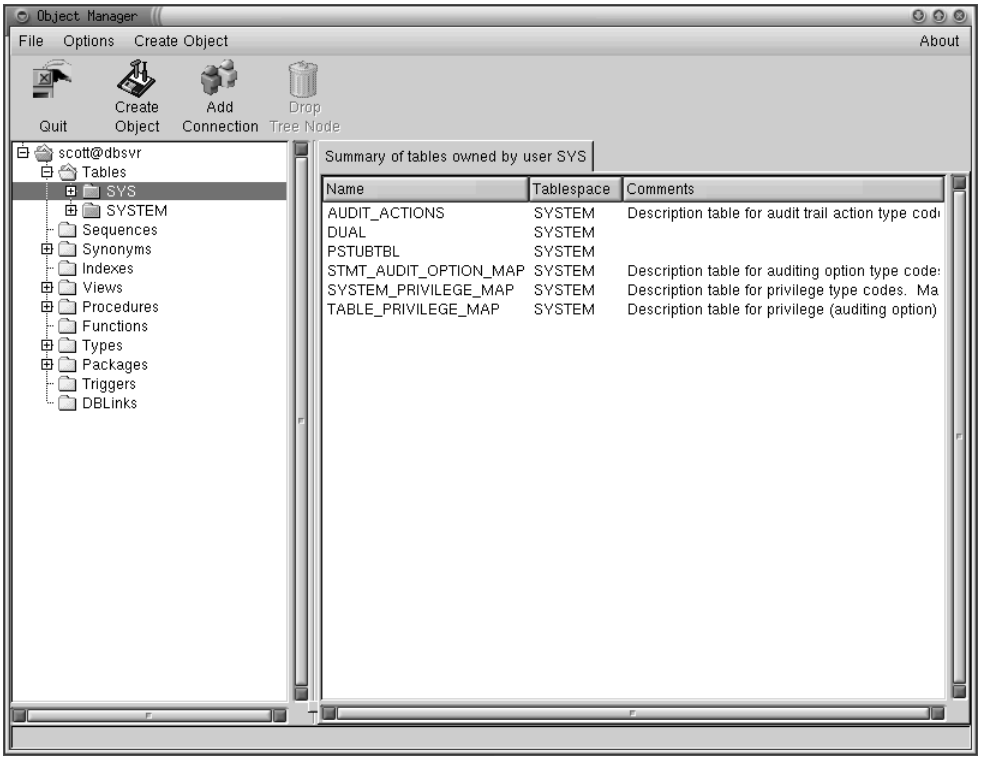

Figure 3–9
ObjectManager de Orasoft

SQLWork

Orasoft SQLWork est une feuille graphique de travail en SQL, qui propose une interface graphique conviviale à Oracle (voir figure 3.8). Écrit à l'aide du GIMP Toolkit (GTK), il est préinstallé sur la plupart des distributions Linux. Il offre beaucoup de possibilités, dont l'historique, les signets, les impressions et l'exportation de résultats dans un fichier. Les résultats peuvent faire l'objet d'interrogations libres (*free-form*) ou en passant par une présentation en grille, similaire à celle d'un tableur. La fonction d'historique dispose d'un buffer pour stocker les commandes précédentes ; la fonction de signet permet de sauvegarder des instructions SQL pour un usage ultérieur, comme on le fait avec un navigateur web. Les résultats des requêtes sont conservés dans un fichier ou imprimés.

© copyright Éditions Eyrolles

ObjectManager

Orasoft ObjectManager (OM) est destiné à l'administration des bases de données. L'écran d'ObjectManager est divisé en deux parties (voir figure 3.9). La partie gauche de l'écran donne une liste de dossiers qui représentent les différents types d'objets disponibles dans le SGBDR Oracle. En réalité, il ne s'agit pas exactement d'objets. Par exemple, il existe un dossier qui contient les connexions courantes à la base de données. On utilise OM pour consulter, supprimer et créer des objets, dont les tables, les index, les synonymes et les vues ; il donne aussi des informations sur les liens db, les procédures, les vues, les paquetages, les déclencheurs, les synonymes, les séquences et les types. La partie droite de la fenêtre affiche des informations relatives à l'objet sélectionné, réparties selon les sujets, accessibles *via* des tabulations.

Orac

Orac (www.kkitts.com/orac_dba/) est un outil d'administration de bases de données conçu pour les utilisateurs dotés des privilèges DBA (voir figure 3.10). L'application, open-source, est écrite en Perl/Tk, un ensemble d'outils écrits initialement pour fournir des fonctions graphiques au langage de script Tcl. Tk est maintenant disponible pour Perl. Orac permet de réaliser des rapports, des graphiques et d'éditer les tables. Il regorge de rapports intéressants. Par exemple, on peut obtenir un rapport complet, avec diagramme à barres, de l'utilisation de l'espace. Il y a aussi un menu de réglage. Il faut un peu de temps pour maîtriser l'interface, mais une fois que vous l'avez en main, elle vous en fait gagner.

Perl/DBI

Perl est devenu un langage de programmation web courant. On obtient une plate-forme Internet puissante en combinant Oracle avec Perl, le serveur web Apache et Linux, et il est facile de faire fonctionner Oracle avec Perl. Se connecter à Oracle avec Perl nécessite deux paquetages, DBI et DBD. DBI est une interface indépendante de base de données : c'est un paquetage de fonctions de base de données indépendant de la plate-forme. Le second composant, DBD, est un pilote de base de données. Chaque serveur de base de données exige un pilote de base de données spécifique — DBD::Oracle est la version utilisée avec Oracle. On télécharge ces deux composants à partir d'un site *Comprehensive Perl Archive Network* (CPAN). La liste des sites CPAN est disponible sur http://www.perl.com/.

© copyright Éditions Eyrolles

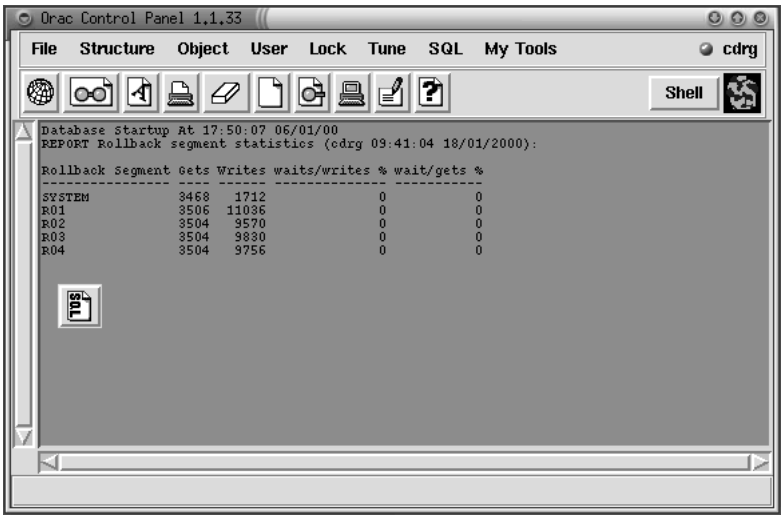
Orac Control Panel 1.1.33
File Structure Object User Lock Tune SQL My Tools cdrg
Shell
Database Startup At 17:50:07 06/01/00
REPORT Rollback segment statistics (cdrg 09:41:04 18/01/2000):
Rollback Segment Gets Writes waits/writes % wait/gets %
SYSTEM 3468 1712 0 0
R01 3506 11036 0 0
R02 3504 9570 0 0
R03 3504 9830 0 0
R04 3504 9756 0 0
SQL

Figure 3–10
Orac

Voici un exemple simple de programme qui vérifie un code de statut et imprime un message en se servant de Perl pour se connecter à Oracle :

```
# !/usr/bin/perl
use DBI;

# Le nom du serveur est Orcl, l'utilisateur webuser, le mot de passe iluv8i

my $DBH = DBI->connect('orcl', 'webuser', 'iluvo8i') or
          warn("Cannot connect to database ", $DBI::errstr);
my $sth = $DBH->prepare("select stat_msg from cust_table
          where stat_code=1002");
$sth->execute;
my $hashref = $sth->fetchrow_hashref;
print "Status Message: $hashref->{STAT_MSG}\n"
$DBI->disconnect ;
```

Résumé

Dans ce chapitre, nous avons passé en revue les concepts clés d'Oracle à comprendre pour travailler avec le SGBDR Oracle. Le plus important est la différence entre une base de données et une instance. La base de données représente les structures physiques, alors que l'instance représente les structures de la mémoire.

© copyright Éditions Eyrolles

La zone globale système (SGA) est un composant mémoire critique : la SGA est un facteur clé dans le réglage des performances.

Les installations de Oracle8 et de Oracle8i exigent des étapes de préinstallation similaires. Oracle8 a besoin de moins de ressources matérielles et a été examiné plus longuement, par contre Oracle8i est orienté vers l'Internet et comprend un environnement d'exécution Java natif, dans le serveur, ainsi que de nouvelles fonctionnalités de sécurité et XML. Oracle8 a été conçu pour Red Hat Linux 5.2 et a besoin de bibliothèques de compatibilité pour s'exécuter sous les versions Red Hat Linux 6.0 et 6.1.

Oracle8 et 8i présentent de nombreuses caractéristiques de sécurité dont les privilèges système, les privilèges d'objets et les rôles. Le dictionnaire des données est un référentiel central, en lecture seule, des tables et des vues. Oracle se sert du dictionnaire des données pour trouver les définitions d'objets, l'allocation de l'espace, les contraintes d'intégrité, les privilèges et les informations de vérification. SQL*Plus est l'interface d'origine du SGBDR Oracle fourni par Oracle Corporation. Les utilitaires Import et Export servent à charger des données dans la base de données ou à en extraire ; SQL*Loader permet d'importer des données d'autres bases de données. SQLWork, Object Manager et Orac sont des exemples d'applications tierces pour Linux. Perl est un langage excellent de développement web et a une excellente connectivité à Oracle.

FAQ

Q. Pourquoi ne puis-je exécuter aucun des utilitaires Oracle ?

R. Assurez-vous que le répertoire **$ORACLE_HOME/bin** figure bien dans votre PATH.

Q. Que représente le mot de passe du compte interne ? Quand j'essaie de me connecter en tant que compte interne dans le gestionnaire de serveur, il me demande un mot de passe.

R. Vous êtes probablement connecté dans un compte qui n'est pas membre du groupe *dba*, à consulter dans le fichier **/etc/group,** en tant que root.

Q. Le programme d'installation Oracle échoue et réclame un programme manquant appelé **gmake**. Pourquoi ?

R. Vérifiez si **gmake** est bien installé sous **/usr/bin**. S'il est installé, assurez-vous que votre PATH comprend le répertoire **/usr/bin**. Si **gmake** n'existe pas, vous pouvez, en tant que root, créer un lien symbolique vers **make** : **ln –s /usr/bin/make /usr/bin/gmake**.

© copyright Éditions Eyrolles

Q. Où se trouve le fichier **libclntsh.so.8.0** ?

R. Les fichiers **libclntsh.so.1.0** pour Oracle8 et **libclntsh.so.8.0** pour Oracle8i peuvent être générés à l'aide du script **genclntsh**, situé dans **$ORACLE_HOME/bin**. En principe, il est créé pendant l'installation. Exécutez **genclntsh** si vous avez des problèmes de connexion entre des machines 8.1.5 et 8.0.5.

Q. Où puis-je trouver plus d'informations sur Oracle ?

R. Le réseau OTN, *Oracle Technology Network* (http://technet.oracle.com) fournit de la documentation sur tous les produits Oracle. L'éditeur O'Reilly (http://www.oreilly.com/) a une excellente collection de livres sur Oracle. *Oracle : The complete reference* par George Koch et Kevin Loney est un livre important, une référence utile.

Q. Où puis-je trouver un groupe d'utilisateurs Oracle ?

R. Le groupe international des utilisateurs Oracle (http://www.ioug.org/) tient la liste des groupes d'utilisateurs Oracle.

© copyright Éditions Eyrolles

4

Installer Informix sur Linux

Dans ce chapitre

- Installation
- Besoins d'espace
- Configuration
- Autres outils
- Ressources

© copyright Éditions Eyrolles

Introduction

Informix, comme Oracle, est un produit de base de données puissant. Informix, Oracle et d'autres bases de données professionnelles telles que Sybase et DB2 de IBM, se concurrencent sur des plates-formes variées, mais elles visent principalement la gestion des bases de données au niveau de l'entreprise, où entrent en jeu de gros volumes, tant pour les transactions sur les bases de données que pour la mémoire. Ces produits sont considérés comme de sérieux concurrents dans les projets de serveurs de base de données de production, où l'on rencontre certaines contraintes de charge propres à l'environnement des sociétés commerciales de taille moyenne à grande.

Informix est l'une des premières bases de données sur UNIX. Informix a été doté du premier ISAM (*Indexed Sequential Access Method*) produit pour UNIX, appelé C-ISAM, encore disponible aujourd'hui et que l'on peut encore trouver pour Linux. Tous les produits de base de données sur le marché fonctionnent avec une méthodologie ISAM, sans exception. Informix a beaucoup inauguré dans le monde des bases de données, et peut maintenant concurrencer nombre de produits plus chers.

Tous les produits de base de données, dont beaucoup ne sont pas cités ici, se heurtent à une contrainte de conformité minimale aux standards ISO et SQL, car ce sont des produits de base de données *d'entrée de gamme*. Ils se conforment tous au point d'entrée minimum de compatibilité SQL de base ; en plus des caractéristiques d'entrée de gamme autorisées par la spécification SQL, ils sont aussi autorisés à proposer des caractéristiques de niveau avancé ou optionnelles. Aucun éditeur n'a implémenté toutes les spécifications SQL, mais la plupart se conforment au minimum requis et choisissent d'implémenter des caractéristiques additionnelles quand ils ont l'opportunité d'augmenter leur compétitivité. Pour plus d'informations sur le standard SQL, rechercher sur le web. Les éditeurs ajoutent aussi des caractéristiques selon les besoins des utilisateurs finals, ou quand la concurrence ajoute une caractéristique, appréciée des utilisateurs, ou dont ils ne peuvent simplement plus se passer.

Ce chapitre est consacré à la manière d'installer et de mettre en œuvre un moteur Informix dans l'environnement d'exploitation Linux. L'accent n'est pas mis seulement sur un moteur de base de données de niveau entreprise, mais aussi sur un moteur adaptable à la progression de Linux vers les plus grands serveurs du marché, surtout dans le marché des systèmes d'exploitation UNIX. Beaucoup de ce qui est abordé ici s'applique aussi aux autres versions UNIX de Informix Dynamic Server dans le reste du monde UNIX et, à un moindre degré, au monde NT. Déjà, rassurez-vous, il n'y a pas à se préoccuper de comparaison avec NT : la plupart des moteurs d'entreprise sont proposés sur UNIX et NT et, dans certains cas, sur MVS (sur mainframes).

© copyright Éditions Eyrolles

Les seuls à ne pas se trouver sur UNIX sont les produits de base de données de Microsoft, qui choisissent de promouvoir NT comme plate-forme étalon. Microsoft ne s'est attaqué que depuis récemment au marché des bases de données, sous la forme de produits de data warehousing et de transformation de données, en complément de sa ligne de produits SQL Server. Sitôt qu'une plate-forme en architecture Intel x86 sera assez grande pour que NT la fasse fonctionner aussi vite qu'une plate-forme UNIX ou autre, SQL Server aura sa chance sur ce terrain. En attendant, vous remarquerez des différences sensibles de performance entre UNIX et NT, et, bien sûr, Linux. NT tend à bien fonctionner en utilisant Informix avec de petits groupes de travail, mais on obtient de meilleurs ratios de performance avec Linux et UNIX.

On préfère d'ordinaire ce que l'on connaît déjà. Si un développeur a affaire à Informix en premier, il sera le plus souvent enclin à donner la préférence à Informix. C'est encore plus vrai quand on voit combien Informix est facile à gérer, comparé à certains des autres produits. La raison en est sa simplicité : une instance Oracle typique demande approximativement 600 Mo d'espace disque, rien que pour le logiciel, sans parler de l'espace supplémentaire nécessaire aux bases de données. Une installation Oracle est bien plus compliquée qu'Informix, non seulement du fait de l'agencement du logiciel, mais aussi de la configuration. Ouvrez le répertoire du logiciel Oracle et vous verrez la différence une fois que vous aurez installé Informix.

Oracle demande beaucoup plus de temps à installer et à configurer qu'Informix, sans parler des performances, comparées à celles que l'on obtient avec Informix. Si Oracle est votre premier moteur, et Informix le second, vous serez surpris de la simplicité de la mise en œuvre. Par contre, si votre premier moteur est Informix, et que vous essayez d'installer le logiciel Oracle, vous pourrez constater que c'est une expérience riche d'enseignements.

Informix prend considérablement moins d'espace qu'Oracle — les outils SDK de développement du noyau et du moteur occupent moins de 150 Mo. Cependant, cela peut varier selon les composants additionnels qu'un développeur ajoute au répertoire du logiciel Informix, connu habituellement dans UNIX sous le nom de **$INFORMIXDIR**. Nous décrirons en détail, dans ce chapitre, la variable d'environnement **$INFORMIXDIR** ainsi que tout ce dont on a besoin pour mettre en œuvre Informix, et le faire fonctionner rapidement et de manière professionnelle.

En principe, une installation Informix peut être terminée en deux heures, voire moins ; cependant, cela dépend un peu de la planification préalable. Plus vous l'avez planifiée en amont, plus l'installation est rapide. Bien que les moteurs Informix soient simples à installer, prenez le temps de comprendre les ramifications de l'organisation que vous mettez en place. Dites-vous bien que c'est vrai de n'importe quel logiciel ou autre produit de base de données.

© copyright Éditions Eyrolles

Nous proposons ici un bon agencement de base sur lequel s'appuyer, qui fonctionnera pour la plupart des moteurs Informix — la seule exception étant le moteur standard, qui n'est pas traité dans ce chapitre, car le moteur standard (SE, *Standard Engine*) ne tire pas vraiment profit du partage de la mémoire partagée, ni de beaucoup d'autres caractéristiques de Informix Dynamic Server.

Le moteur standard est intéressant dans les petits déploiements, qu'il s'agisse de sites ou de logiciels, où les contraintes opérationnelles exigent peu ou aucune intervention humaine, ou bien quand les contraintes des données sont très faibles ou simples, ainsi que les caractéristiques de la base de données. SE fut le moteur de choix pour un grand nombre de marchés verticaux (VAR, *value-added resellers*) qui y ont trouvé un moteur simple, à faible maintenance et à bas prix, adapté à des applications UNIX variées. Même si SE était à l'aise dans des environnements peu coûteux pendant presque toute la vie d'Informix, il n'est pas question que son faible coût continue de tenir lieu de critère unique. Il manque trop de choses à SE pour en faire un choix judicieux par rapport à IDS dans le marché actuel, sauf à ne rechercher que les caractéristiques de base pour une base de données.

C-ISAM se situe encore en dessous ; c'est le niveau le plus bas des produits de base de données que propose Informix. C-ISAM est la base de la plupart des produits de bases de données actuellement sur le marché, y compris les concurrents. Cependant, C-ISAM est extrêmement limité et a une fenêtre d'utilisation étroite, ce qui l'exclut des options envisagées dans la plupart des situations. Garder à l'esprit que Informix a été la première société du marché des bases de données à offrir un produit ISAM pour UNIX, puis qu'elle en est venue à offrir des produits ISAM plus sophistiqués avec son moteur standard, qui permettait l'utilisation du standard SQL pour accéder à la base de données. C-ISAM est disponible sur Linux, et on peut en faire une copie pour apprendre comment il fonctionne, et comment les autres produits de bases de données l'utilisent.

Informix Dynamic Server (IDS), appelé à l'origine *Online Dynamic Server*, tire son nom de la manière dont il fonctionne. IDS effectue de lui-même, de manière dynamique, les ajustements nécessaires lors de son chargement, et offre des options que l'on peut changer sans avoir à redémarrer le moteur. Il y a, bien sûr, des limites mais, pour l'essentiel, ce moteur peut accepter une charge importante et affiche encore des performances impressionnantes. IDS est très fiable et robuste, considérons-le comme le moteur de base, autant que possible. À l'implémentation, il présente à peu près la même simplicité que SE, et ne demande pas beaucoup plus de maintenance que ce que l'on a connu avec le moteur standard. IDS peut être maintenu simplement, pour peu que l'on tienne compte du cycle d'opérations que ce moteur entraîne.

© copyright Éditions Eyrolles

IDS présente, par rapport à SE, deux avantages : des performances plus rapides, mais aussi les types de données avancés. Vous trouverez que IDS ressemble à SE par beaucoup d'aspects, mais qu'il est beaucoup plus puissant et obtient des performances bien meilleures que le moteur standard. Les gains de performance, ainsi que les caractéristiques avancées, en font un choix préférable. Si vous êtes développeur et cherchez autre chose que les performances, vous pouvez aussi tirer bénéfice de la version IDS.2000, qui propose les caractéristiques objet-relationnel, non disponibles dans IDS 7.30. Installer d'abord IDS 7.30 à titre d'exercice pour acquérir une bonne connaissance du moteur de base, puis migrer vers IDS.2000. Il y a des caractéristiques supplémentaires dans IDS.2000 dont vous n'aurez peut-être jamais besoin mais, d'un autre côté, elles peuvent être essentielles. Quelle que soit la vitesse, il vaut mieux se familiariser d'abord avec le plus simple des deux moteurs, puis migrer vers le plus compliqué.

Pour une comparaison complète des produit SE, IDS et IDS.2000, rendez visite au site web d'Informix. En raison des limitations du SE, les sites les plus sérieux retiendront pour la plupart Dynamic Server comme le moteur minimum parmi ceux qui sont offerts, voire tous. Ce n'est que depuis peu que le choix s'est révélé délicat, du fait que le coût d'IDS a sensiblement chuté, ce qui est dû en particulier à l'émergence de Linux, et à la concurrence entre SQL Server et d'autres produits orientés petits serveurs. Linux a été confronté à cette situation depuis quelques mois, car le noyau s'est enrichi au point que Linux peut être considéré sérieusement comme un serveur de base de données. Les avantages de Linux sur NT continuent à être évidents en eux-mêmes ; cependant, même avec Linux, on doit prendre soin de configurer le serveur correctement pour obtenir des performances optimales. Un bon administrateur se consacrera aux travaux préparatoires avant l'installation, et non après.

Le moteur le plus récent et le plus important chez Informix n'est pas IDS 7.30, mais plutôt IDS.2000, un moteur complet, objet-relationnel, qui va jusqu'à offrir des possibilités d'aide à la décision, destinés à des sites qui nécessitent des produits orientés objet avancés de gestion de bases de données relationnelle (SGBDR) compatibles SQL, que IDS ne peut fournir. Nous ne traiterons pas d'IDS.2000 dans ce chapitre, mais gardez à l'esprit que, pour l'essentiel, IDS.2000 fonctionne de la même manière que IDS 7.30. Ce que vous apprenez en IDS 7.30 pourra être appliqué directement à IDS.2000, tout simplement parce que le moteur est le même à la base et que la technologie sophistiquée objet-relationnelle y a été incorporée. Aussi trivial que cela paraisse, il faut comprendre que, bien que ces deux moteurs soient sensiblement différents, leur comportement de base est en grande partie le même. Ce n'est que lorsqu'on utilise les caractéristiques du système de gestion de base de données objet-relationnelle (ORDBMS) que les moteurs se comportent différemment.

© copyright Éditions Eyrolles

Pour la petite histoire : Postgres, un autre moteur de base de données bien connu pour Linux, trouve son origine dans Ingres, notamment chez Michael Stonebreaker, qui a inventé Ingres. Quand Informix a acheté la version commerciale de Postgres, appelée Illustra, la société a surtout acheté les cerveaux concepteurs d'Ingres (maintenant Postgres, en format objet-relationnel), et a renommé le produit Universal Server.

Postgres continue à vivre de son côté, et Universal Server, maintenant absorbé par Informix, a trouvé une suite dans IDS.2000. Les moteurs IDS 7.30 et l'ancien Universal Server 9.x ont été combinés pour donner naissance au meilleur de ces deux mondes dans IDS.2000. Du point de vue de l'architecture, le résultat final est suffisamment semblable au précédent (c'est-à-dire, IDS 7.30) pour permettre la migration d'IDS 7.30 vers IDS.2000, sans aucune — ou presque — modification de l'environnement. Nous ne testerons pas ce cas ici, mais Informix a publié cela comme quelque chose de non seulement faisable, mais aussi comme faisant partie d'une offre promotionnelle d'amélioration.

Les installations d'Informix sont, en principe, divisées en trois parties :

- extraction du logiciel ;
- marquage (*branding*) de l'installation ;
- mise en œuvre du logiciel.

La première étape, l'extraction du logiciel, implique l'acquisition et la mise en place du logiciel. Tout dépend de la forme sous laquelle le logiciel est distribué, et des étapes à parcourir pour mettre le logiciel dans un format qui convient à l'étape suivante, le marquage de l'installation. Ces préliminaires sont souvent les plus difficiles ; le marquage de l'installation proprement dit est la partie la moins pénible de l'installation.

La seconde étape, marquage de l'installation, a quelque peu évolué depuis sa forme d'origine, excepté dans le monde Windows, où InstallShield a pris la place de l'installation UNIX. Informix distribue traditionnellement ses logiciels selon cette méthode en trois étapes, le média de distribution étant en général un CD-ROM contenant un fichier ou des fichiers tar et, dans beaucoup de cas, un CD avec un fichier CPIO. Il y a, bien sûr, des exceptions et d'autres formats. Depuis ces derniers mois, certains des logiciels sont fournis avec une option d'installation d'une interface X Window qui ressemble à la manière dont WordPerfect est installé sur UNIX, mais ce n'est pas une norme.

La distribution Linux des moteurs Informix a fait son apparition, au format du gestionnaire de paquet Red Hat (RPM) et tar, quand le premier portage Linux a été réalisé. Le dernier moteur 7.30, IDS 7.30.UC10, est encapsulé dans un fichier RPM, qui, à son tour, est encapsulé dans un fichier zip. D'autres composants viennent compléter le logiciel, dans des fichiers .tar.gz, entre

© copyright Éditions Eyrolles

autres formats. Dans notre exemple d'installation, nous travaillerons avec des fichiers zip et RPM. L'outil de compression Linux **unzip** sera nécessaire — s'assurer d'avoir installé cet outil ou quelque autre outil qui comprenne le format de fichier compressé zip — à ne pas confondre avec le format gzip.

L'étape finale, rendre le logiciel opérationnel, dépend beaucoup du type de matériel dont vous disposez, et de celui que vous voulez utiliser avec le logiciel. Dans la plupart des cas, nous concevons une base de données comme le principal occupant d'une plate-forme à base de serveur.

De nouveau, nous avons choisi de nous concentrer sur les serveurs Linux, lors de notre installation. Les serveurs Linux sont parmi les plus courants, du fait de la souplesse de cet environnement d'exploitation, sans parler du coût extrêmement bas de l'ajout de ressources, dans les petits comme dans les grands environnements. Une fraction, de jour en jour plus large, de la population informatique se tourne vers Linux, et il se peut que cela représente une bonne carte à jouer, à l'avenir.

Installation

Nous débattrons de l'installation en trois parties : acquisition et mise en place du logiciel, installation et marquage, et initialisation du disque et opérations sur le serveur.

Acquisition et mise en place du logiciel sur le serveur

Il faut initialiser trois variables d'environnement dans les comptes informix et root. Oui, il faut un id informix et un groupe informix. Nous verrons cela plus tard, mais c'est une bonne idée de se poser la question dès maintenant. Pour l'instant, créons ces variables d'environnement sur le système ; nous aurons l'occasion d'y revenir :

```
INFORMIXDIR=/opt/informix/ids  ; export INFORMIXDIR
INFORMIXSERVER=rebel_shm ; export INFORMIXSERVER
ONCONFIG=rebel_shm.cfg ; export ONCONFIG
```

Remplacer le nom *rebel* par le vôtre ; il servira de référence partout. Bien placer ces commandes dans les profils **informix** et **root**, et s'assurer que les profils mettent bien ces valeurs dans l'environnement à la connexion, avant de commencer l'installation. Tant que les variables **$INFORMIXDIR** ne sont pas initialisées, l'installation ne peut pas se poursuivre. Déconnectez et reconnectez-vous, et assurez-vous que ces variables sont bien initialisées.

© copyright Éditions Eyrolles

Télécharger le fichier zip depuis www.intraware.com. Dans ce fichier zip se trouve un fichier RPM qui contient des fichiers tar et un script d'installation. Bien suivre les recommandations d'Informix pour l'installation de leur logiciel.

Important. Les étapes suivantes seront menées sous le login **root**. Si vous n'avez pas accès à root, arrêtez-vous et demandez la permission d'utiliser le login root pour la durée de l'installation. Une fois que le logiciel a été installé, vous aurez encore besoin du login root pour initialiser les répertoires de données. N'utilisez pas un autre login sauf si la documentation d'installation Informix le préconise, ou vous signale qu'un login non root est admis. Après l'installation et la configuration, vous pouvez vous passer du login root, et utiliser le login **informix** pour l'administration courante.

Double-cliquer sur le fichier zip pour l'ouvrir dans Karchiver si celui-ci a été installé, et si l'on se sert de l'interface de navigation de fichier HTML/Web de KDE comme gestionnaire de fichier. C'est plus facile qu'il n'y paraît — il suffit de double-cliquer sur le fichier zip dans une fenêtre d'auto-démarrage, et Karchiver affiche la liste de son contenu. À ce moment, extraire tout ce que l'on voit et le mettre dans un répertoire de travail, mais pas dans celui où l'on va installer en définitive le logiciel. Dans le menu Fichier, choisir l'option **Extract to**, choisir un répertoire et extraire tous les fichiers (figure 4.1).

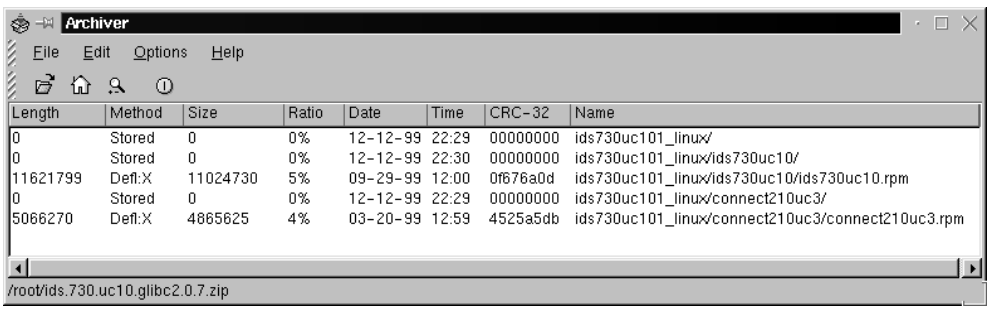
Archiver
File Edit Options Help
Length Method Size Ratio Date Time CRC-32 Name
0 Stored 0 0% 12-12-99 22:29 00000000 ids730uc101_linux/
0 Stored 0 0% 12-12-99 22:30 00000000 ids730uc101_linux/ids730uc10/
11621799 Defl:X 11024730 5% 09-29-99 12:00 0f676a0d ids730uc101_linux/ids730uc10/ids730uc10.rpm
0 Stored 0 0% 12-12-99 22:29 00000000 ids730uc101_linux/connect210uc3/
5066270 Defl:X 4865625 4% 03-20-99 12:59 4525a5db ids730uc101_linux/connect210uc3/connect210uc3.rpm
/root/ids.730.uc10.glibc2.0.7.zip

Figure 4–1
Vue du fichier zip à télécharger, dans Karchiver

Une fois tous les fichiers extraits dans un répertoire, on peut aussi voir le contenu du fichier RPM, avec **kpackage** ; s'assurer qu'il est installé aussi sur la machine. **kpackage** ne fait pas encore partie des paquetages d'installation de KDE standard, mais figure dans les versions alpha/bêta des programmes KDE (figure 4.2).

Notez à la figure 4.2 que ces fichiers sont extraits directement dans le répertoire ***/root.*** Assurez-vous que vous avez assez de place pour conserver les fichiers extraits. Les deux fichiers RPM à extraire contiennent approximativement 60 Mo de logiciel. Ils peuvent être placés n'importe où pour l'instant,

© copyright Éditions Eyrolles

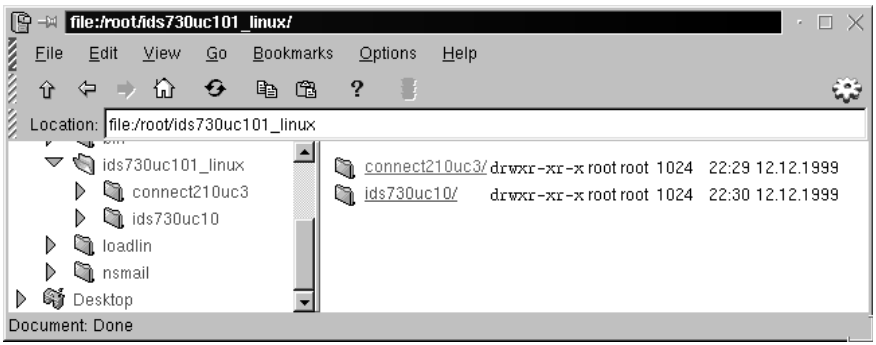

Figure 4–2
Vue des répertoires créés après extraction des fichiers et d'un RPM du client SDK

parce qu'ils ne sont utilisés que pendant l'installation ; après quoi, on peut les supprimer ou les ranger ailleurs, pour libérer un peu d'espace sur le disque. On peut aussi les graver sur un CD-ROM.

Pour décompresser le contenu du fichier RPM, utiliser soit **kpackage**, soit les commandes **rpm** stipulées dans les documents d'installation d'Informix. À ce point, garder à l'esprit que si l'on utilise **kpackage** pour installer le logiciel, il ne pourra l'installer que dans **/opt/informix** : il faudra un répertoire **/opt** pour ce faire, sinon le programme **kpackage** échouera. Le mieux est d'utiliser la commande **rpm** sur la ligne de commande, parce que l'on peut indiquer explicitement où mettre le logiciel, en s'appuyant sur la variable $**INFORMIXDIR**.

Maintenant, il est grand temps d'indiquer, dans cette variable d'environnement, l'endroit où mettre le logiciel Informix. L'initialiser dans le login root, dans lequel on aura déjà créé un id **informix** et un groupe **informix**. Utiliser la commande d'administration du système adéquate pour créer ce nouvel id utilisateur, et suivre la méthode recommandée par la distribution Linux pour ajouter les utilisateurs et les groupes. La plupart des distributions récentes proposent un programme pour cela, et il vaut mieux s'en servir, plutôt que d'éditer les fichiers **/etc/passwd** et **/etc/group,** même si c'était la manière de procéder des premières distributions de Linux et d'UNIX.

Il est recommandé de suivre les conseils d'Informix, et d'utiliser les commandes **rpm** en ligne pour faciliter l'installation ; cela facilite la besogne s'il se produit un incident. Les versions récentes d'Informix proposent une assistance technique limitée quand on achète le logiciel, et il n'y a aucune raison de ne pas suivre les instructions d'installation qui sont fournies.

On peut consulter le site http://www.informix.com/informix/resource/install/linux/0730x/01.fm.html, une documentation en ligne qui donne les instructions, étape par étape, pour installer le logiciel. Penser, à cette étape de l'installation, que nous venons seulement de trouver un lieu où installer le

© copyright Éditions Eyrolles

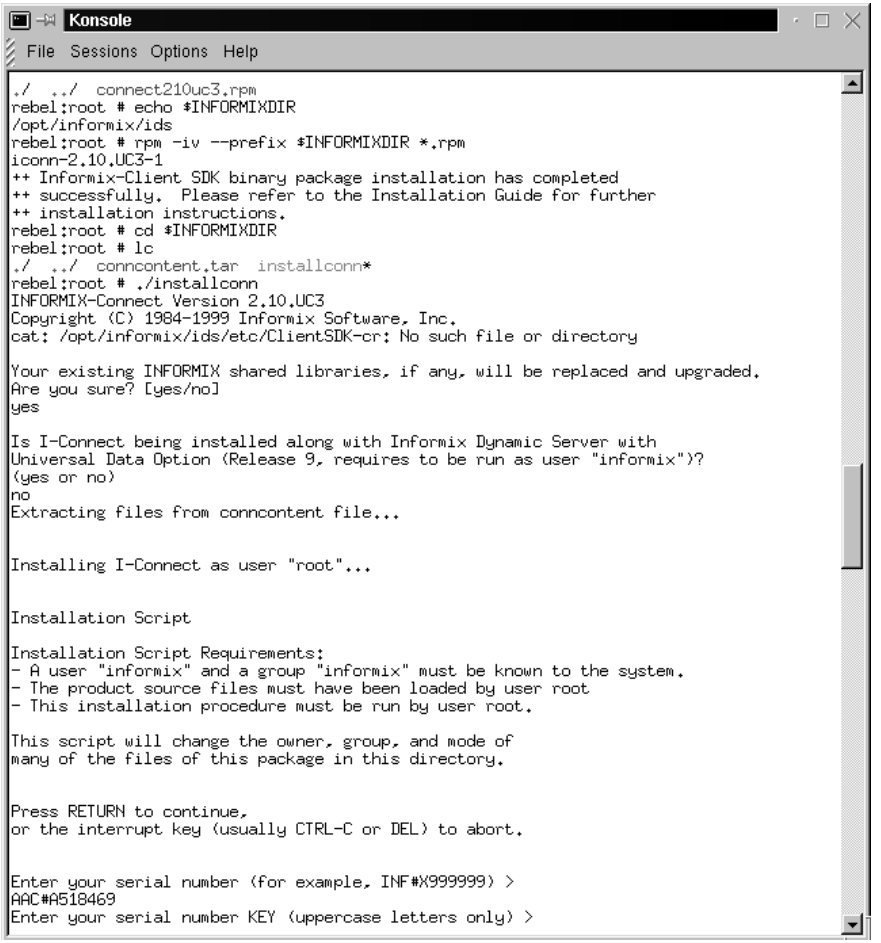
Konsole
File Sessions Options Help
./ ../ connect210uc3.rpm
rebel:root # echo $INFORMIXDIR
/opt/informix/ids
rebel:root # rpm -iv --prefix $INFORMIXDIR *.rpm
iconn-2.10.UC3-1
++ Informix-Client SDK binary package installation has completed
++ successfully. Please refer to the Installation Guide for further
++ installation instructions.
rebel:root # cd $INFORMIXDIR
rebel:root # lc
./ ../ conncontent.tar installconn*
rebel:root # ./installconn
INFORMIX-Connect Version 2.10.UC3
Copyright (C) 1984-1999 Informix Software, Inc.
cat: /opt/informix/ids/etc/ClientSDK-cr: No such file or directory
Your existing INFORMIX shared libraries, if any, will be replaced and upgraded.
Are you sure? [yes/no]
yes
Is I-Connect being installed along with Informix Dynamic Server with
Universal Data Option (Release 9, requires to be run as user "informix")?
(yes or no)
no
Extracting files from conncontent file...
Installing I-Connect as user "root"...
Installation Script
Installation Script Requirements:
- A user "informix" and a group "informix" must be known to the system.
- The product source files must have been loaded by user root
- This installation procedure must be run by user root.
This script will change the owner, group, and mode of
many of the files of this package in this directory.
Press RETURN to continue,
or the interrupt key (usually CTRL-C or DEL) to abort.
Enter your serial number (for example, INF#X999999) >
AAC#A518469
Enter your serial number KEY (uppercase letters only) >

Figure 4–3
Démarrer l'installation du client SDK (Iconnect)

logiciel. Notre prochaine étape sera la phase d'installation et de marquage. Quand on a trouvé un endroit où placer le logiciel et qu'il a été extrait du fichier RPM, on peut le marquer et configurer l'espace de base de données du disque pour conserver les bases de données et les tables.

Installation et marquage

Créer un répertoire **/opt/informix/ids** si vous avez un répertoire **/opt.** Sinon, créer un répertoire **informix** sous le répertoire où l'on veut installer le logiciel. D'habitude, je crée un répertoire **/opt/informix/ids** sous celui-ci, de sorte que,

© copyright Éditions Eyrolles

s'il faut installer d'autres moteurs, on puisse créer des répertoires séparés pour ces autres logiciels et conserver celui qui est déjà installé. Il faut donc avoir un répertoire **/opt/informix/ids** pour faire l'installation. On peut l'appeler comme on veut, à condition d'indiquer le nom du répertoire dans la variable **INFORMIXDIR**. Dans notre cas, cela donne :

```
INFORMIXDIR=/opt/informix/ids  ; export INFORMIXDIR
```

Il faut le faire avant d'utiliser les commandes **rpm** en ligne de commande, de manière à extraire le logiciel en se référant à **$INFORMIXDIR**. Le signe $ qui figure au début des lignes de commandes est simplement une invite ; ne pas l'utiliser dans les commandes citées.

```
$ cd /root/ids730uc101_linux/connect210uc3
$ rpm –iv –prefix $INFORMIXDIR *.rpm
$ cd $INFORMIXDIR
$. /installconn
```

Quand le script d'installation du client SDK est terminé (figure 4.3), exécuter les mêmes opérations pour installer le serveur :

```
$ cd /root/ids730uc101_linux/ids730uc10
$ rpm –iv –prefix $INFORMIXDIR *.rpm
$ cd $INFORMIXDIR
$. /installserver
```

On peut installer soit le SDK soit le moteur en premier — cela n'a pas d'importance. Extraire le logiciel du **rpm**, marquer le logiciel, extraire le second logiciel du second **rpm**, puis marquer. C'est la méthode correcte qui vous évitera des problèmes. Une fois achevée cette phase de l'installation, vous êtes prêt à initialiser le serveur pour le rendre opérationnel (figure 4.4).

Il est important de relancer le script d'installation de chaque produit si l'on transfère l'arborescence des répertoires sur un autre système ou sur un autre lecteur. Il suffit d'exécuter le script d'installation de chaque produit, en s'assurant au préalable que $**INFORMIXDIR** est bien renseigné, et la réinstallation se fait sans problème. Remarque : il y a quelques opérations de dernière minute (établir des liens) qui doivent être faites dans l'ordre pour que le serveur fonctionne correctement. L'installation exécute cette étape à la fin ; il est donc important de réinstaller si l'on transfère le répertoire ou l'arbre. Il faut se souvenir que Linux n'a aucun registre, contrairement à Windows NT ou 98, de sorte que les seules choses dont on doit se préoccuper sont les liens finaux créés à la fin de l'installation. On peut aussi visiter le site web d'Informix pour trouver des astuces et des techniques de récupération de l'espace disque occupé par certains fichiers NLS, installés pour des langages dont on n'a pas besoin.

© copyright Éditions Eyrolles

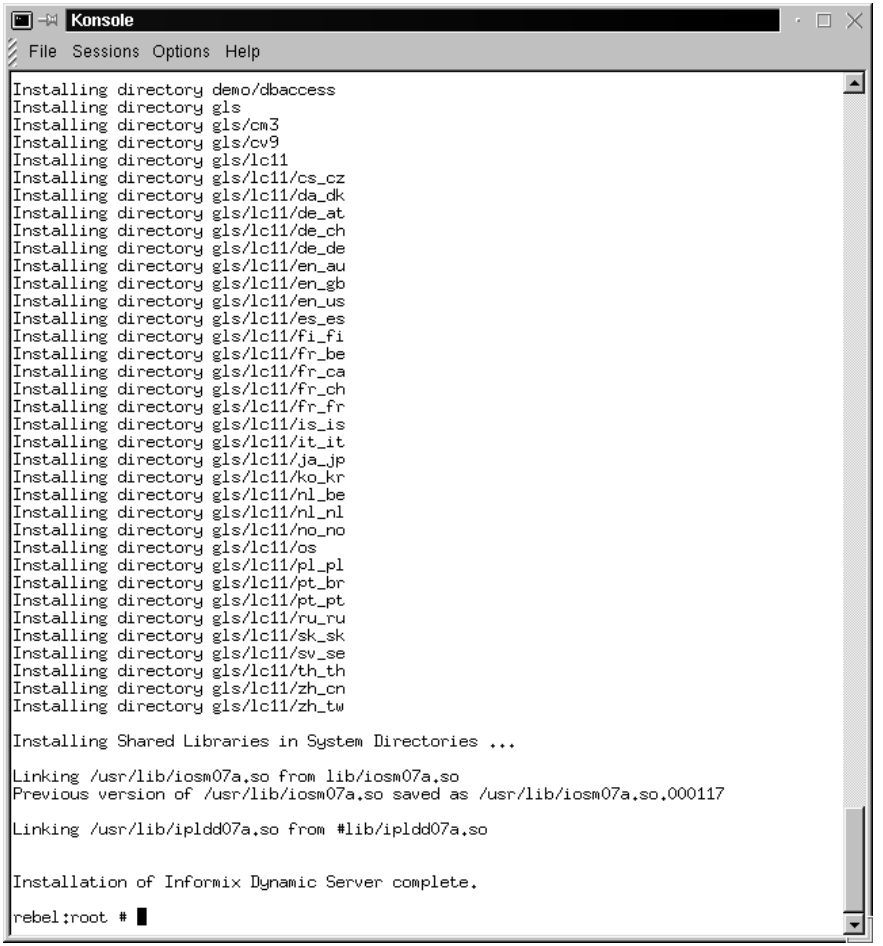

Figure 4–4
Achèvement avec succès de l'installation d'un serveur IDS

On peut trouver un script qui supprime certains fichiers, mais il faut savoir que la réinstallation échoue, dans ce cas, si ces fichiers n'existent pas. Il faudra extraire à nouveau le logiciel, puis refaire l'installation de base à partir du répertoire temporaire.

Un dernier point sur cette partie de l'installation : n'enlevez pas le fichier tar de **iconnect** tant que vous n'êtes pas certain de ne plus devoir le réinstaller. Il faut espérer que le programme d'installation du SDK ne cherche pas spécialement le fichier tar dans ce répertoire précis, car cela demande deux fois plus d'espace — uniquement pour installer le SDK. À l'avenir, Informix permettra de mettre ailleurs sur le système les fichiers tar et les scripts d'installation, en vue des réinstallations, ce qui réduira l'allocation d'espace pour le logiciel.

© copyright Éditions Eyrolles

Initialisation du disque et démarrage du serveur

Nous pouvons maintenant aborder les aspects opérationnels d'IDS. Ce dont il va être question doit être préparé à l'avance, et un certain volume d'espace disque réservé pour les *dbspaces*, où résideront la base de données réelle et les tables. Ce chapitre met l'accent sur les aspects de l'installation d'un serveur Informix : il s'agit de se familiariser avec une certaine terminologie d'Informix avant de poursuivre.

Parcourir le *Guide de l'administrateur pour IDS*, que l'on trouve sur le site web d'Informix ou sur CD-ROM. Une impression papier de ce manuel est disponible, mais à un certain prix. Un conseil : trouver le dernier CD-ROM de la documentation connue sous le titre *Documentation en ligne et réponses*, qui contient des documents pour tous les produits de base de données Informix. Le CD-ROM de cette documentation est disponible directement chez Informix, ou peut être commandé auprès du bureau de ventes régional. La documentation est encore disponible sur le site web Informix : prendre le temps de la télécharger et se familiariser avec le *Guide de l'administrateur* ; ce manuel décrit tout ce que vous avez besoin de savoir pour que votre installation se passe bien.

Abordons maintenant les avantages respectifs des fichiers bruts non formatés (*raw disk*) et des fichiers préformatés, prêts à l'emploi (*cooked*) qu'il vaut mieux bien connaître. Actuellement, Linux n'est pas encore vraiment orienté disque brut comme le sont beaucoup de plates-formes UNIX, mais il est probable que cette approche sera mise en place au moment de la publication de cet ouvrage. Certains sites se servent de disques bruts sans incident, mais, actuellement, cette option n'est pas prise en charge.

Une installation *raw disk*, en bref, permet de lier logiquement le logiciel Informix en direct sur des disques non formatés (des lecteurs physiques qui ne sont pas montés par le système de fichier) ; c'est le logiciel Informix qui prend en charge les opérations de formatage et d'accès aux disques. Avec les disques bruts, Informix joue le rôle de système d'exploitation disque pour les lecteurs initialisés en tant que *raw disk*. La seule chose qui reste à la charge de Linux est de fournir une entrée dans le répertoire **/dev** qui pointe vers le lecteur de disques. Un lien logique est créé dans un répertoire que l'on choisit, pour pointer vers le répertoire **/dev** et l'entrée du lecteur. Par exemple, créons un répertoire **/opt/informix/links**, puis un lien dans ce répertoire qui pointe vers le répertoire **/dev.** S'il y a un disque dans **/dev** appelé **/dev/rdisk1**, il suffit de créer le lien comme ceci :

```
$ cd /opt/informix/links
$ ln –s /dev/rdisk1 datadbs01
```

Quand on fait un **ls –l**, on voit le lien, mais pas le disque.

© copyright Éditions Eyrolles

Une installation de disques préformatés (*cooked*), par contre, permet de créer des fichiers Linux d'une certaine taille, que Linux gérera en même temps qu'Informix. Les fichiers cooked sont des fichiers Linux normaux et se présentent comme tels. Les fichiers cooked sont créés avec une taille spécifique explicite et font l'objet d'une sauvegarde standard, sous réserve que le serveur Informix soit hors ligne et arrêté. Les fichiers cooked doivent être fermés, comme tout fichier Linux, pour pouvoir être sauvegardés. L'avantage des fichiers cooked est simplement que ce sont des fichiers Linux normaux et que leur sauvegarde est aussi simple que celle des autres fichiers Linux.

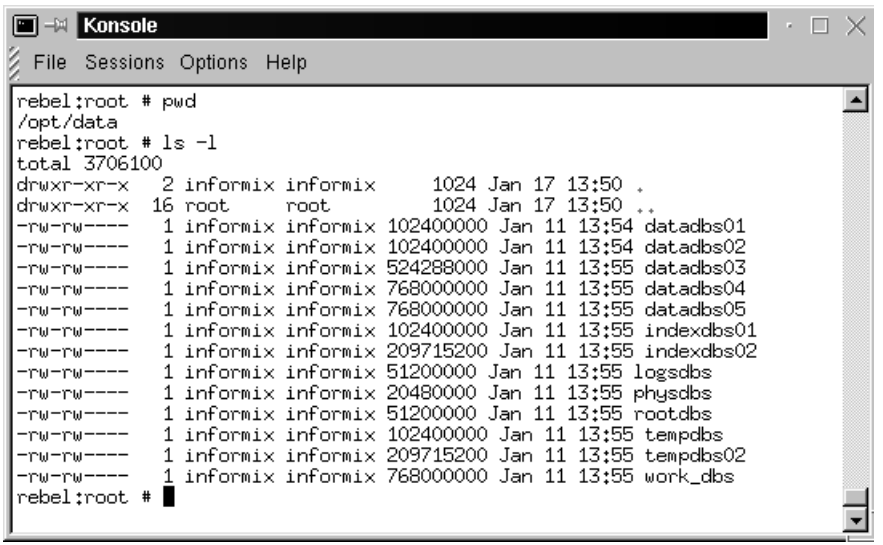
Konsole
File Sessions Options Help
rebel:root # pwd
/opt/data
rebel:root # ls -l
total 3706100
drwxr-xr-x 2 informix informix 1024 Jan 17 13:50 .
drwxr-xr-x 16 root root 1024 Jan 17 13:50 ..
-rw-rw---- 1 informix informix 102400000 Jan 11 13:54 datadbs01
-rw-rw---- 1 informix informix 102400000 Jan 11 13:54 datadbs02
-rw-rw---- 1 informix informix 524288000 Jan 11 13:55 datadbs03
-rw-rw---- 1 informix informix 768000000 Jan 11 13:55 datadbs04
-rw-rw---- 1 informix informix 768000000 Jan 11 13:55 datadbs05
-rw-rw---- 1 informix informix 102400000 Jan 11 13:55 indexdbs01
-rw-rw---- 1 informix informix 209715200 Jan 11 13:55 indexdbs02
-rw-rw---- 1 informix informix 51200000 Jan 11 13:55 logsdbs
-rw-rw---- 1 informix informix 20480000 Jan 11 13:55 physdbs
-rw-rw---- 1 informix informix 51200000 Jan 11 13:55 rootdbs
-rw-rw---- 1 informix informix 102400000 Jan 11 13:55 tempdbs
-rw-rw---- 1 informix informix 209715200 Jan 11 13:55 tempdbs02
-rw-rw---- 1 informix informix 768000000 Jan 11 13:55 work_dbs
rebel:root #

Figure 4–5
Exemple de listing du répertoire de fichiers cooked dbspaces dont se sert le logiciel Informix pour ranger les données

Quand on exécute un **ls –l** sur le répertoire, les fichiers cooked apparaissent comme des fichiers Linux normaux (voir figure 4.5). Quand on inspecte un disque brut, le listing du répertoire montre un lien vers **/dev entries**, mais aucune taille de fichier n'apparaît, car ces disques sont sous le contrôle d'Informix. Pour en savoir plus sur les disques bruts, utiliser la commande Informix **onstat** qui montre l'espace utilisé ; c'est le seul moyen de voir ce qu'ils contiennent. Utiliser les fichiers cooked tant que l'on n'est pas familiarisé avec l'initialisation des disques bruts.

Informix sur Linux obtient de très bonnes performances avec les fichiers cooked, mais il est préférable, sur UNIX, d'utiliser les disques bruts : les performances sont souvent de 15 à 20 % supérieures à celles des fichiers cooked. Actuellement, on ne dispose pas encore de bancs d'essai qui compa-

© copyright Éditions Eyrolles

rent les différences entre disques bruts et fichiers cooked sur Linux. Beaucoup de sites x86, en version NT de Informix Dynamic Server, se servent exclusivement de fichiers cooked, simplement parce que les disques bruts sur NT n'offrent pas des gains de performance suffisants pour compenser les risques à les utiliser. Le *Guide de l'administrateur* étudie cette question en détail.

Les fichiers cooked sont plus faciles à gérer dans la plupart des situations ; il est conseillé de les utiliser dans les environnements où l'on veut une administration simple et où l'on n'envisage pas une charge extrêmement lourde sur le système. Si l'on a vraiment besoin de performances accrues, on aura la plupart du temps intérêt à migrer vers une installation de disques bruts sur UNIX. Tant que les disques bruts ne sont pas supportés sur Linux, utiliser les fichiers cooked.

Création des fichiers de données pour Informix

Créer un second répertoire, **/opt/data**, ou **/db/data**, ou, de préférence sur un lecteur physique séparé, un répertoire appelé **/data**, dans lequel on crée un sous-répertoire appelé **/data/informix**. Soit un lecteur séparé appelé **/db**, et un répertoire **/db/data**, qui est la racine de toutes les données. Pour notre installation, nous utiliserons **/opt/data**, mais vous aurez un répertoire différent. Maintenir, autant que possible, les données séparées du logiciel. Comme nous le verrons plus loin, il faut allouer au moins 500 Mo pour les données, sinon plus. Faire des économies de bouts de chandelle sur l'espace des données n'a aucun sens, sauf si on ne peut l'éviter. Lorsque nous parlons d'allocation d'espace, cela veut dire l'espace total, pas celui de fichiers spécifiques. L'espace total agrégé ne sera pas inférieur à 500 Mo, et nous créerons des fichiers cooked de différentes tailles dans cet espace.

Informix se sert de *chunks* (morceaux) pour décrire la valeur la plus grande de l'espace des données. Toutes les données résident dans les *chunks* à l'intérieur de pages, qui sont la plus petite unité de mesure de l'espace des données. Le lecteur se reportera au *Guide de l'administrateur* pour une description complète des *chunks* et des pages. Comme ce chapitre utilise cette terminologie pour expliquer la fin de l'installation, il est important de comprendre ces concepts. Avec le logiciel Informix, vous trouverez une excellente documentation, alors profitez-en.

Après s'être familiarisé avec les concepts de *chunks* et de *dbspaces*, on peut configurer le serveur pour le démarrer. Initialisons un serveur qui permet à chacun de spécifier sa propre manière d'utiliser l'espace des données ainsi que l'espace opérationnel minimal à mettre en réserve. Cette méthode vous permettra d'augmenter la taille du système avec un minimum de désagréments.

© copyright Éditions Eyrolles

Il faut réserver de l'espace disque pour un certain nombre de *dbspaces*. Il s'agit d'un minimum. Pour le reste, c'est-à-dire les espaces disques à ajouter, c'est à vous d'en évaluer les besoins. Voici les espaces opérationnels :

- **Rootdbs :** contient la base de données sysmaster, le centre intelligent du moteur de base de données.
- **Physdbs :** sert à stocker les données avant de les écrire sur le disque. Il est compris dans le rootdbs, mais le transférer dans son propre dbspace améliore les performances et l'administration. On peut l'appeler comme on veut : cependant, il est recommandé de le nommer de manière cohérente avec sa fonction.
- **Logsdbs :** contient les logs logiques. Après l'installation, il est inclus dans le rootdbs, mais il sera placé dans son propre dbspace, pour améliorer le fonctionnement. On peut l'appeler comme on veut ; néanmoins, il est recommandé de le nommer lui aussi de manière cohérente.
- **Tempdbs01** : sert aux tables temporaires et au tri des requêtes, que l'on nomme aussi comme on veut, de manière cohérente...

Les suivants sont des espaces de données, que l'on nomme comme on veut.

- **Dbs_dbs** : dédié à la création de la base de données, mais non aux tables. Une entrée de base de données est créée, ainsi que quelques entrées de tables système de base, mais les tables utilisateur sont référencées en dehors de cet espace.
- **Datadbs01 :** tables utilisateur.
- **Indexdbs01 :** index utilisateur.

Si vous créez cet ensemble minimal de *dbspaces*, vous aurez une installation minimale qui peut convenir à la plupart de vos besoins, dans le cadre de l'initiation au fonctionnement du moteur IDS. Il vous permettra aussi des extensions faciles, en cas de besoin.

À propos des disques

Initialiser une zone, sur le système, pour recevoir Informix. Il y a trois zones à prendre en considération : le logiciel, les dbspaces et une zone de travail pour les sauvegardes, les chargements et autres tâches.

Les disques SCSI sont recommandés, de préférence aux disques IDE, en raison de leur vitesse et de leur souplesse. SCSI permet de chaîner ensemble des lecteurs plus facilement qu'IDE. Si l'on a une expérience suffisante, on peut mettre au point, sur IDE, une initialisation de disque similaire, sachant qu'il y a encore des limitations sur IDE. SCSI est donc plus souple. Aussi, n'hésitez pas à équiper votre système d'un contrôleur et de lecteurs SCSI.

© copyright Éditions Eyrolles

Disques et répertoires

Voici la description du contenu des répertoires **/opt/informix/ids7.30** et **/opt/data** :

- **/opt/informix/ids7.30** contient le logiciel — tout le logiciel est situé dans cet arbre. Réserver au minimum 500 Mo pour le logiciel Informix. Aucune donnée ne doit être rangée ici. Se servir de cette convention pour le nom de répertoire : cela permet d'ajouter d'autres versions du logiciel sous **/opt/informix** et de tester facilement la migration ou la mise à jour du logiciel, par un simple changement de la variable d'environnement **$INFORMIXDIR**.
- **/opt/data** contient les dbspaces des fichiers cooked ; si possible, le mettre sur un autre disque. Ce répertoire est réservé aux données.

Besoins d'espace

Combien d'espace faut-il réserver ? On peut prendre un disque de 2 Go et allouer l'espace comme suit :

- pour les espaces opérationnels :
  - 50 Mo pour les **rootdbs** ;
  - 50 Mo pour le **physdbs** ;
  - 100 Mo pour **logsdbs** ;
  - 300 Mo pour **tempdbs** ;

  soit 500 Mo au total ;
- pour les espaces de données :
  - 100 Mo pour **dbs_dbs** — il contient seulement les entrées de base de données et ne sert pas pour les tables ;
  - 250 Mo pour **datadbs01** — il peut être plus petit si vous ne l'utilisez que pour la formation ; il contiendra des tables ;
  - 250 Mo pour **indexdbs01** — là encore, il peut être plus petit pour la formation ; il ne contiendra que des index ;

  soit 600 Mo au total.

Si vous avez réservé un disque de 2 Go, par exemple, vous avez maintenant alloué un total de 1 100 Mo pour démarrer et il reste encore de l'espace pour des besoins futurs. Ce n'est qu'un point de départ, et pas nécessairement la

© copyright Éditions Eyrolles

panacée pour tous les systèmes. Il s'agit aussi d'options qui préservent les ressources, sans gaspillage d'espace pour chaque dbspace. Les seules options de ressources vraiment obligatoires sont celles de rootdbs et de physdbs, mais même celles-ci sont arbitraires : après avoir correctement configuré le système, vous trouverez les tailles adaptées à votre site. Démarrer avec celles-ci, et les changer plus tard si besoin est. L'espace des rootdbs ne servira plus guère désormais, aussi il est inutile de l'agrandir au-delà de 50 Mo ; dans la plupart des cas, 20 Mo sont même trop. Je n'alloue 50 Mo que par précaution.

/work est un système de fichier monté à la demande pour l'espace de travail. Réserver au moins 1 000 Mo pour ce disque. C'est un disque Linux qui sert à préparer les fichiers de chargement et à gérer les données exportées ou importées dans la base de données. Il est toujours utile d'avoir un espace de travail sur le système, spécifique de différents projets de base de données. Sous ce répertoire, vous pouvez créer des répertoires de projet ou des répertoires individuels, à votre convenance.

Important. Beaucoup de BIOS de PC ne permettent pas l'utilisation de grands lecteurs dans Linux, même si le matériel est de facture récente. La plupart des BIOS ne supportent aucun des plus grands lecteurs — faire attention quand on se sert des lecteurs IDE. C'est l'une des raisons les plus importantes d'utiliser les SCSI : on n'est pas limité par le contrôleur interne pour l'agencement des disques quand on configure les lecteurs pour Linux.

Cela n'est pas la seule raison d'utiliser des lecteurs SCSI ; la performance en est une autre. Malgré les progrès des lecteurs IDE, la plupart des sites continuent à configurer des lecteurs SCSI et à leur donner la préférence. Si vous avez déjà configuré votre lecteur, réservez un volume d'espace important pour le logiciel et les données. Vous êtes prêt à installer Informix.

Configuration

Maintenant que nous avons vu comment allouer l'espace, nous allons configurer le serveur pour sa toute première opération en ligne.

Comme indiqué au début de ce chapitre, il y a trois variables d'environnement à initialiser. Nous pouvons maintenant travailler avec la seconde variable, **INFORMIXSERVER**, et la troisième, **ONCONFIG**.

Dans le cadre de ce chapitre, nous nous intéressons seulement à la mise en œuvre du serveur, et non aux considérations sur les performances, qui pourront être abordées lors d'un processus d'apprentissage, qu'il s'agisse des essais ou des erreurs, à l'aide de la documentation opérationnelle d'Informix.

© copyright Éditions Eyrolles

La variable **INFORMIXSERVER** correspond directement au **DBSERVERNAME** dans le fichier **$ONCONFIG.** Utiliser, pour **DBSERVERNAME**, le même nom que celui qui a servi à initialiser **INFORMIXSERVER**. Cela est essentiel pour la connexion réseau entre un client et le serveur. On utilisera, pour la variable **INFORMIXSERVER**, le même nom sur les clients que l'on souhaite connecter à ce serveur. **INFORMIXSERVER** sert aussi aux utilitaires, tels que **onstat**, pour communiquer avec l'instance correcte du moteur.

$INFORMIXDIR/etc/$ONCONFIG

Aller dans le répertoire **$INFORMIXDIR**, et parcourir l'arbre des répertoires.

```
/ont/informix/I.D.S.
                    |__aaodir
                    |__bin

                    |__dbssodir
                    |__demo
                        |__dbaccess
                    |__etc

                    |__forms
                    |__gls
                        |__cm3
                        |__cv9
                        |__lc11
                            |__cs_cz
                            |__da_dk
                            |__de_at
                            |__de_ch
                            |__de_de
                            |__en_au
                            |__en_gb
                            |__en_us
                            |__es_es
                            |__fi_fi
                            |__fr_be
                            |__fr_ca
                            |__fr_ch
                            |__fr_fr
                            |__is_is
                            |__it_it
                            |__ja_jp
                            |__ko_kr
                            |__nl_be
                            |__nl_nl
                            |__no_no
                            |__os
```

© copyright Éditions Eyrolles

```
                |__pl_pl
                |__pt_br
                |__pt_pt
                |__ru_ru
                |__sk_sk
                |__sv_se
                |__th_th
                |__zh_cn
                |__zh_tw
|__incl
    |__esql
    |__hpl

|__lib
    |__cli
    |__dmi
    |__esql

|__msg
    |__en_us
        |__0333
|__release
    |__en_us
        |__0333
```

Nous allons travailler dans **$INFORMIXDIR** et dans le répertoire Linux **/etc** normal. Comme vous pouvez le voir, le logiciel Informix a son propre répertoire /**etc**, où se trouvent tous les fichiers nécessaires à la configuration. Les deux plus critiques, **onconfig.std** et **sqlhosts**, feront l'objet de notre attention avant l'installation. Faire des copies avant de lancer le moteur, car après l'installation, il faudra encore travailler sur le **$ONCONFIG** et y faire les ajustements nécessaires pour rendre IDS complètement opérationnel.

En plus de ce que vous lirez ici ou dans le *Guide de l'administrateur*, il faudra consulter les notes de la version IDS qui contiennent des instructions d'installation de dernière minute : elles se trouvent dans le répertoire **$INFORMIXDIR/release/en_us/0333.** Il y a des notes importantes concernant tous les aspects du logiciel installé.

Aller dans le répertoire **$INFORMIXDIR/etc** et passer la commande **ls –CaF**. On voit apparaître toutes sortes de fichiers — entre autres, **onconfig.std**. Il faut faire une copie de ce fichier — la copie portera le nom indiqué dans la variable **$ONCONFIG**.

```
$ cd $INFORMIXDIR/etc
$ cp onconfig.std rebel_shm.cfg
```

Dans notre exemple, il s'appelle **rebel_shm.cfg**. Le **_shm** à la fin est une sorte de convention : on peut avoir plusieurs instances différentes d'Informix qui

© copyright Éditions Eyrolles

fonctionnent sur le même serveur ; **_shm** signifie simplement *shared memory* (mémoire partagée), et relève d'un goût personnel. Là aussi, appelez le fichier **ONCONFIG** comme vous voulez — assurez-vous seulement de le nommer correctement, comme nous l'avons montré.

Une fois que vous l'avez copié, mettez-vous dans votre éditeur de texte favori, et suivez les instructions qui suivent. Il est important de sauvegarder ce fichier en format texte ASCII, aussi assurez-vous que votre éditeur autorise l'édition de texte simple. Personnellement, je préfère utiliser vi ; mais, dans Linux, il y a plusieurs éditeurs de texte, qui ressemblent à Notepad de Microsoft Windows.

Remplacer, dans la commande **vi rebel_shm.cfg**, **vi** par le nom de l'éditeur de texte, puis éditer le fichier **ONCONFIG**.

```
$ vi rebel_shm.cfg

#**************************************************************
#
#                          INFORMIX SOFTWARE, INC.
#
#  Title:       onconfig.std
#  Description: Informix Dynamic Server Configuration Parameters
#
#**************************************************************

# Root Dbspace Configuration

ROOTNAME rootdbs           # Root dbspace name
ROOTPATH /dev/online_root  # Path for device containing root dbspace
ROOTOFFSET 0               # Offset of root dbspace into device (Kbytes)
ROOTSIZE 20000             # Size of root dbspace (Kbytes)

# Disk Mirroring Configuration Parameters

MIRROR 0                   # Mirroring flag (Yes = 1, No = 0)
MIRRORPATH                 # Path for device containing mirrored root
MIRROROFFSET 0             # Offset into mirrored device (Kbytes)

# Physical Log Configuration

PHYSDBS rootdbs            # Location (dbspace) of physical log
PHYSFILE 1000              # Physical log file size (Kbytes)


# Logical Log Configuration

LOGFILES 6                 # Number of logical log files
LOGSIZE 500                # Logical log size (Kbytes)
```

© copyright Éditions Eyrolles

```
# Diagnostics

MSGPATH /usr/informix/online.log        # System message log file path
CONSOLE /dev/console                    # System console message path
ALARMPROGRAM /usr/informix/etc/log_full.sh   # Alarm program path
SYSALARMPROGRAM /usr/informix/etc/evidence.sh
                          # System Alarm program path
TBLSPACE_STATS 1

# System Archive Tape Device

TAPEDEV /dev/tapedev      # Tape device path
TAPEBLK 16                # Tape block size (Kbytes)
TAPESIZE 10240            # Maximum amount of data to put on tape (Kbytes)

# Log Archive Tape Device

LTAPEDEV /dev/tapedev     # Log tape device path
LTAPEBLK 16               # Log tape block size (Kbytes)
LTAPESIZE 10240           # Max amount of data to put on log tape (Kbytes)

# Optical

STAGEBLOB                 # Informix Dynamic Server/Optical staging area

# System Configuration

SERVERNUM 0               # Unique id corresponding to a
                          # Dynamic Server instance
DBSERVERNAME              # Name of default database server
DBSERVERALIASES           # List of alternate dbservernames
NETTYPE                   # Configure poll thread(s) for nettype
DEADLOCK_TIMEOUT 60       # Max time to wait of lock in distributed env.
RESIDENT 0                # Forced residency flag (Yes = 1, No = 0)

MULTIPROCESSOR 0          # 0 for single-processor,
                          # 1 for multi-processor
NUMCPUVPS 1               # Number of user (cpu) vps
SINGLE_CPU_VP 0           # If non-zero, limit number of cpu vps to one

NOAGE 0                   # Process aging
AFF_SPROC 0               # Affinity start processor
AFF_NPROCS 0              # Affinity number of processors

# Shared Memory Parameters

LOCKS 2000                # Maximum number of locks
BUFFERS 200               # Maximum number of shared buffers
NUMAIOVPS                 # Number of IO vps
PHYSBUFF 32               # Physical log buffer size (Kbytes)
LOGBUFF 32                # Logical log buffer size (Kbytes)
LOGSMAX 6                 # Maximum number of logical log files
CLEANERS 1                # Number of buffer cleaner processes
SHMBASE 0x10000000L       # Shared memory base address
```

© copyright Éditions Eyrolles

```
SHMVIRTSIZE 8000          # initial virtual shared memory segment size
SHMADD 8192               # Size of new shared memory segments (Kbytes)
SHMTOTAL 0                # Total shared memory (Kbytes). 0=>unlimited
CKPTINTVL 300             # Check point interval (in sec)
LRUS 8                    # Number of LRU queues
LRU_MAX_DIRTY 60          # LRU percent dirty begin cleaning limit
LRU_MIN_DIRTY 50          # LRU percent dirty end cleaning limit
LTXHWM 50                 # Long transaction high water mark percentage
LTXEHWM 60                # Long transaction high water mark (exclusive)
TXTIMEOUT 300             # Transaction timeout (in sec)
STACKSIZE 32              # Stack size (Kbytes)

# System Page Size
# BUFFSIZE - Dynamic Server no longer supports this configuration
# parameter. To determine the page size used by Dynamic Server on your
# platform see the last line of output from the command, 'onstat -b'.

# Recovery Variables
# OFF_RECVRY_THREADS : Number of parallel worker threads
# during fast recovery or an offline restore.
# ON_RECVRY_THREADS : Number of parallel worker threads
# during an online restore.

OFF_RECVRY_THREADS 10     # Default number of offline worker threads
ON_RECVRY_THREADS 1       # Default number of online worker threads

# Data Replication Variables
# DRAUTO: 0 manual, 1 retain type, 2 reverse type
DRAUTO 0                  # DR automatic switchover
DRINTERVAL 30             # DR max time between DR buffer flushes
                          # (in sec)
DRTIMEOUT 30              # DR network timeout (in sec)
DRLOSTFOUND /usr/informix/etc/dr.lostfound  # DR lost+found file path

# CDR Variables
CDR_LOGBUFFERS 2048       # size of log reading buffer pool (Kbytes)
CDR_EVALTHREADS 1,2       # evaluator threads (per-cpu-vp,additional)
CDR_DSLOCKWAIT 5          # DS lockwait timeout (seconds)
CDR_QUEUEMEM 4096         # Maximum amount of memory for any CDR queue
                          # (Kbytes)

# Backup/Restore variables
BAR_ACT_LOG /tmp/bar_act.log
BAR_MAX_BACKUP 0
BAR_RETRY 1
BAR_NB_XPORT_COUNT 10
BAR_XFER_BUF_SIZE 31

# Informix Storage Manager variables
ISM_DATA_POOL ISMData     # If the data pool name is changed, be sure to
                          # update $INFORMIXDIR/bin/onbar. Change to :
                          # ism_catalog create_bootstrap -pool <new name>
ISM_LOG_POOL ISMLogs
```

© copyright Éditions Eyrolles

```
# Read Ahead Variables
RA_PAGES                  # Number of pages to attempt to read ahead
RA_THRESHOLD              # Number of pages left before next group

# DBSPACETEMP:
# Dynamic Server equivalent of DBTEMP for SE. This is the list of
# dbspaces that the Dynamic Server SQL Engine will use to create temp
# tables etc. If specified it must be a colon separated list of
# dbspaces that exist when the Dynamic Server system is brought
# online.  If not specified, or if all dbspaces specified are
# invalid, various ad hoc queries will create temporary files in
# /tmp instead.

DBSPACETEMP               # Default temp dbspaces

# DUMP*:
# The following parameters control the type of diagnostics information
# which is preserved when an unanticipated error condition (assertion
# failure) occurs during Dynamic Server operations.
# For DUMPSHMEM, DUMPGCORE and DUMPCORE 1 means Yes, Ø means No.

DUMPDIR /tmp              # Preserve diagnostics in this directory
DUMPSHMEM 1               # Dump a copy of shared memory
DUMPGCORE Ø               # Dump a core image using 'gcore'
DUMPCORE Ø                # Dump a core image
                          # (Warning:this aborts Dynamic Server)
DUMPCNT 1                 # Number of shared memory or gcore dumps for
                          # a single user's session

FILLFACTOR 9Ø             # Fill factor for building indexes
                          # method for Dynamic Server to use when
                          # determining current time
USEOSTIME Ø               # Ø: use internal time(fast),
                          # 1: get time from OS(slow)

# Parallel Database Queries (pdq)

MAX_PDQPRIORITY 1ØØ       # Maximum allowed pdqpriority
DS_MAX_QUERIES            # Maximum number of decision support queries
DS_TOTAL_MEMORY           # Decision support memory (Kbytes)
DS_MAX_SCANS 1Ø48576      # Maximum number of decision support scans
DATASKIP                  # List of dbspaces to skip

# OPTCOMPIND
# Ø => Nested loop joins will be preferred (where
#      possible) over sortmerge joins and hash joins.
# 1 => If the transaction isolation mode is not
#      "repeatable read", optimizer behaves as in (2)
#      below.  Otherwise it behaves as in (Ø) above.
# 2 => Use costs regardless of the transaction isolation
#      mode.  Nested loop joins are not necessarily
#      preferred.  Optimizer bases its decision purely
#      on costs.
```

© copyright Éditions Eyrolles

```
OPTCOMPIND 2              # To hint the optimizer

ONDBSPACEDOWN 2
# Dbspace down option: 0 = CONTINUE, 1 = ABORT, 2 = WAIT
LBU_PRESERVE 0            # Preserve last log for log backup
OPCACHEMAX 0              # Maximum optical cache size (Kbytes)

# HETERO_COMMIT (Gateway participation in distributed transactions)
# 1 => Heterogeneous Commit is enabled
# 0 (or any other value) => Heterogeneous Commit is disabled
HETERO_COMMIT 0

# Optimization goal: -1 = ALL_ROWS(Default), 0 = FIRST_ROWS
OPT_GOAL        -1

# Optimizer DIRECTIVES ON (1/Default) or OFF (0)
DIRECTIVES 1

# Status of restartable restore
RESTARTABLE_RESTORE OFF
```

La première section est extrêmement importante, parce qu'il faut maintenant changer les options pour indiquer où seront rangées les données. Dans notre exemple, nous utiliserons le répertoire **/opt/data** pour y mettre les données, y compris les rootdbs.

```
# Root Dbspace Configuration - avant

ROOTNAME rootdbs          # Root dbspace name
ROOTPATH /dev/online_root # Path for device containing root dbspace
ROOTOFFSET 0              # Offset of root dbspace into device (Kbytes)
ROOTSIZE 20000            # Size of root dbspace (Kbytes)
```

Maintenant, effectuons la modification (ligne ROOTPATH) :

```
# Root Dbspace Configuration - après

ROOTNAME rootdbs          # Root dbspace name
ROOTPATH /opt/data/rootdbs # Path for device containing root dbspace
ROOTOFFSET 0              # Offset of root dbspace into device (Kbytes)
ROOTSIZE 20000            # Size of root dbspace (Kbytes)
```

Notre système se servira d'un fichier cooked pour les rootdbs ; nous avons mis à jour cette variable. Rappelons que la taille de rootdbs peut être fixée à 50 Mo au lieu de 20 Mo. On peut l'arrondir à 50 000 Ko, ce qui est presque la même chose ; modifiez la taille maintenant si vous le souhaitez (ligne ROOTSIZE), sinon laissez la valeur par défaut de 20 000 Ko. Ce n'est qu'une précaution. Personnellement, je donne un peu plus d'espace aux rootdbs pour des raisons de sécurité ; se souvenir qu'après configuration, il aura un usage réel très faible, sauf pour le moteur. Dans certains cas, le choix d'un rootdbs

© copyright Éditions Eyrolles

légèrement plus grand est judicieux : par exemple, si on laisse les logs logiques ou le log physique dans les rootdbs.

Important. Si l'on envisage de créer plus d'une instance sur le serveur, renommer rootdbs en rootdbs01, physdbs en physdbs01, logsdbs en logsdbs01 et tempdbs en tempdbs01 si l'on doit se référer à ces dbspaces dans l'installation. De cette manière, les noms sont cohérents avec les instances et sont plus faciles à gérer. Faire les changements appropriés dans le fichier **$ONCONFIG** de ce serveur avant d'exécuter **oninit –i**. Ne pas changer les options de **PHYSDBS** avant le premier **oninit**, quand on exécute **oninit–i**. Faire ce changement après le démarrage du moteur.

Quand on installe le moteur, le log physique et les logs logiques sont situés dans les rootdbs ; nous allons les transférer dans leurs propres espaces après l'installation. De ce fait, les rootdbs seront très peu utilisés et il vaut mieux ne pas leur donner une taille importante qui serait un gaspillage d'espace disque. La seule raison de changer cela maintenant est que l'on ne peut pas le faire plus tard, sauf si l'on réinitialise l'instance. Il faut donc décider maintenant si l'on change la taille, ou si l'on vivra toujours avec la taille de 20 Mo par défaut. Si on laisse 20 000 Ko, s'assurer de créer un logsdbs et un physdbs pour les logs logiques et le log physique, en suivant les instructions, données plus loin dans ce chapitre, pour les transférer dans leurs propres dbspaces. Après déplacement des logs, il y aura approximativement 16 Mo inutilisés sur les 20 Mo. Ce transfert est conseillé, à titre de pratique standard, qui donne, plus tard, une souplesse totale quand il faut ajuster le nombre et la taille des logs logiques ou la taille du log physique. Nous exécuterons cette étape à la fin, quand le moteur sera en fonctionnement.

Poursuivons la configuration de notre serveur. Nous pourrons avoir ou non un système de bande pour la sauvegarde mais, pour l'instant, désactivons la sauvegarde sur bande, jusqu'à ce que le serveur soit totalement opérationnel.

```
# System Archive Tape Device - avant

TAPEDEV /dev/tapedev       # Tape device path
TAPEBLK 16                 # Tape block size (Kbytes)
TAPESIZE 10240             # Maximum amount of data to put on tape (Kbytes)

# Log Archive Tape Device

LTAPEDEV /dev/tapedev      # Log tape device path
LTAPEBLK 16                # Log tape block size (Kbytes)
LTAPESIZE 10240            # Max amount of data to put on log tape (Kbytes)
```

Nous faisons nos changements comme suit :

```
# System Archive Tape Device - après

# TAPEDEV /dev/tapedev     # Tape device path
TAPEDEV /dev/null          # Tape device path
```

© copyright Éditions Eyrolles

```
TAPEBLK 16                  # Tape block size (Kbytes)
TAPESIZE 10240              # Max amount of data to put on log tape (Kbytes)
# Log Archive Tape Device

# LTAPEDEV /dev/tapedev     # Log tape device path
LTAPEDEV /dev/null          # Log tape device path
LTAPEBLK 16                 # Log tape block size (Kbytes)
LTAPESIZE 10240             # Max amount of data to put on log tape (Kbytes)
```

Les valeurs d'origine sont laissées en commentaires de sorte qu'il est facile de revenir en arrière plus tard et de les changer, et qu'elles servent de référence. Il est toujours bon de mettre les valeurs non retenues en commentaire — copier et changer la copie plutôt que changer l'entrée d'origine. De cette manière, on garde trace des changements.

Le jeu d'unités de bande reste à **/dev/null**, ce qui autorise le serveur IDS à fonctionner sans se soucier de l'endroit où ira la sauvegarde — pour l'instant. Plus tard, nous changerons cela pour une sauvegarde sur bande si nous en avons une, ou nous le laisserons tel quel jusqu'à ce que nous décidions d'avoir une réelle sauvegarde sur bande. **/dev/null** présente un double intérêt : d'une part, signifier au serveur IDS d'ignorer le périphérique de sauvegarde, et d'autre part, faire le nécessaire pour que le moteur soit au courant d'éventuelles sauvegardes, notamment celle des logs logiques. Nous verrons les logs logiques plus tard ; pour l'instant, préoccupons-nous seulement de démarrer le serveur.

Passons maintenant à la section des diagnostics, et faisons là aussi quelques changements.

```
# Diagnostics  - avant

MSGPATH /usr/informix/online.log
                            # System message log file path
CONSOLE /dev/console        # System console message path
ALARMPROGRAM /usr/informix/etc/log_full.sh
                            # Alarm program path
SYSALARMPROGRAM /usr/informix/etc/evidence.sh
                            # System Alarm program path
TBLSPACE_STATS 1
```

Créer un répertoire séparé de logs pour les messages que le moteur émettra sur son fonctionnement. Les valeurs par défaut doivent être modifiées : créer simplement un répertoire maintenant, et autoriser le online.log et les messages de la console à se placer dans ce nouveau répertoire :

```
$ cd $INFORMIXDIR
$ mkdir logs
```

© copyright Éditions Eyrolles

```
$ chown informix:informix logs
$ chmod 775 logs

# Diagnostics - après

MSGPATH /opt/informix/ids/logs/online.log
                        # System message log file path
# CONSOLE /dev/null
                        # System console message path
CONSOLE /opt/informix/ids/logs/console.log
                        # Pour notre installation initiale seulement !
ALARMPROGRAM /opt/informix/ids/etc/log_full.sh
                        # Alarm program path
SYSALARMPROGRAM /opt/informix/ids/etc/evidence.sh
                        # System Alarm program path
TBLSPACE_STATS 1
```

Remarquez que les messages sont envoyés en sortie de la **CONSOLE** vers **/dev/null** et **/opt/informix/ids/logs/console.log**. Il vaut mieux, pour la procédure d'installation initiale, mettre **/opt/informix/ids/logs/console.log**, pour conserver les messages d'installation, mais on peut toujours modifier cela plus tard. La plupart des messages de la console n'apportent pas d'informations intéressantes ; là encore, c'est une option à gérer soi-même : on peut souhaiter les voir tous dans un premier temps, et préférer, par la suite, ne les voir que s'ils sont importants.

Passons à la section de configuration du système, où nous opérerons aussi quelques modifications :

```
# System Configuration - avant

SERVERNUM 0             # Unique id corresponding to a Dynamic Server
                        # instance
DBSERVERNAME            # Name of default database server
DBSERVERALIASES         # List of alternate dbservernames
NETTYPE                 # Configure poll thread(s) for nettype
DEADLOCK_TIMEOUT 60     # Max time to wait of lock in distributed env.
RESIDENT 0              # Forced residency flag (Yes = 1, No = 0)
```

C'est ici que nous donnons un nom à l'instance du moteur et que nous lui indiquons comment communiquer avec le réseau :

```
# System Configuration – après

SERVERNUM 0             # Unique id corresponding to a Dynamic Server
                        # instance
DBSERVERNAME rebel_shm  # Name of default database server
DBSERVERALIASES rebel_net  # List of alternate dbservernames
```

© copyright Éditions Eyrolles

```
NETTYPE soctcp,1,1,NET      # Configure poll thread(s) for nettype
DEADLOCK_TIMEOUT 60         # Max time to wait of lock in distributed env.
RESIDENT 0                  # Forced residency flag (Yes = 1, No = 0)
```

Garder trace de ces noms ; remplacer la partie **rebel** par le nom déjà fixé. Nous nous référerons à ces options dans le fichier **$INFORMIXDIR/etc/sqlhosts,** c'est pourquoi il faut les retenir.

Il y a une mise à jour à faire concernant les logs logiques avant de démarrer le serveur, qui évite de relancer le moteur plus tard. Si l'on a adopté le processus de transfert des logs logiques en dehors des rootdbs, changer la valeur par défaut de **LOGSMAX** — qui démarre à 6 — en une valeur plus élevée (mise en gras dans l'exemple de code qui suit). On trouve cette valeur dans la section mémoire partagée du fichier **$INFORMIXDIR/etc/$ONCONFIG** :

```
# Shared Memory Parameters

LOCKS 2000                  # Maximum number of locks
BUFFERS 200                 # Maximum number of shared buffers
NUMAIOVPS                   # Number of IO vps
PHYSBUFF 32                 # Physical log buffer size (Kbytes)
LOGBUFF 32                  # Logical log buffer size (Kbytes)
LOGSMAX 6                   # Maximum number of logical log files
CLEANERS 1                  # Number of buffer cleaner processes
SHMBASE 0x10000000L         # Shared memory base address
SHMVIRTSIZE 8000            # initial virtual shared memory segment size
SHMADD 8192                 # Size of new shared memory segments (Kbytes)
SHMTOTAL 0                  # Total shared memory (Kbytes). 0=>unlimited
CKPTINTVL 300               # Check point interval (in sec)
LRUS 8                      # Number of LRU queues
LRU_MAX_DIRTY 60            # LRU percent dirty begin cleaning limit
LRU_MIN_DIRTY 50            # LRU percent dirty end cleaning limit
LTXHWM 50                   # Long transaction high water mark percentage
LTXEHWM 60                  # Long transaction high water mark (exclusive)
TXTIMEOUT 300               # Transaction timeout (in sec)
STACKSIZE 32                # Stack size (Kbytes)
```

La changer en 50 :

```
LOGSMAX 50                  # Maximum number of logical log files
```

Cela permet d'ajouter plus tard des logs logiques sans avoir à relancer le moteur. Nous effectuerons cette manipulation au cours de l'installation, aussi changeons cela dès maintenant.

Il est important de laisser toutes les autres options à leur valeur par défaut jusqu'à ce que le moteur soit démarré et fonctionne. En procédant aux ajustements par étapes, nous minimisons les risques d'échec au démarrage.

© copyright Éditions Eyrolles

Il existe d'autres changements possibles, que nous ferons, et à l'avenir vous pourrez les faire, avec un peu d'expérience ; pour l'instant, laissons les valeurs par défaut.

Certaines options seront initialisées pour la réplication des données ; bien que cela ne nous concerne pas à ce stade, il est bon de modifier le chemin, pour le rendre conforme à l'installation :

```
# Data Replication Variables - avant
# DRAUTO: 0 manual, 1 retain type, 2 reverse type
DRAUTO 0                    # DR automatic switchover
DRINTERVAL 30               # DR max time between DR buffer flushes (in sec)
DRTIMEOUT 30                # DR network timeout (in sec)
DRLOSTFOUND /usr/informix/etc/dr.lostfound # DR lost+found file path
```

Après nos changements :

```
# Data Replication Variables - après
# DRAUTO: 0 manual, 1 retain type, 2 reverse type
DRAUTO 0                    # DR automatic switchover
DRINTERVAL 30               # DR max time between DR buffer flushes (in sec)
DRTIMEOUT 30                # DR network timeout (in sec)
DRLOSTFOUND /opt/informix/ids/etc/dr.lostfound    # DR lost+found file path
```

Passons ensuite à la section de sauvegarde, un peu plus loin, pour une autre modification :

```
# Backup/Restore variables  - avant
BAR_ACT_LOG /tmp/bar_act.log
BAR_MAX_BACKUP 0
BAR_RETRY 1
BAR_NB_XPORT_COUNT 10
BAR_XFER_BUF_SIZE 31
```

Ici, nous devons seulement indiquer notre répertoire, **$INFORMIXDIR/logs** :

```
# Backup/Restore variables – après
BAR_ACT_LOG /opt/informix/ids/logs/bar_act.log
BAR_MAX_BACKUP 0
BAR_RETRY 1
BAR_NB_XPORT_COUNT 10
BAR_XFER_BUF_SIZE 31
```

Ce fichier peut devenir assez gros et il faudra gérer sa croissance. On peut créer un script de vidage du fichier log, ce que nous verrons à la fin de ce chapitre. Nous ferons un vidage non seulement de ce fichier mais aussi de online.log, ce qui permet de conserver des fichiers log petits, et convient bien à une analyse périodique. On les archive chaque jour, chaque semaine ou chaque mois, selon les besoins.

© copyright Éditions Eyrolles

Ici se termine l'édition du fichier **ONCONFIG.** Sauvegardez ce fichier avec les changements effectués et préparez-vous à éditer un autre fichier important, le fichier **$INFORMIXDIR/etc/sqlhosts.** Nous ferons plus tard d'autres changements dans le fichier **$ONCONFIG**, mais pour l'instant c'est tout ce qu'il faut faire pour démarrer le moteur.

Les tâches suivantes concernent l'édition d'autres fichiers ASCII. Assurez-vous que ces fichiers sont sauvegardés en tant que fichiers texte, comme nous l'avons fait pour **ONCONFIG**.

$INFORMIXDIR/etc/sqlhosts

Ajouter ces entrées au fichier **$INFORMIXDIR/etc/sqlhosts** :

```
rebel_shm     onipcshm     rebel     sv_rebel_shm
rebel_net     onsoctcp     rebel     sv_rebel_net
```

Elles se rapportent directement aux autres entrées du fichier **/etc/services,** aussi gardez-en la trace. La signification de ces valeurs est exposée dans le *Guide de l'administrateur.* En matière de convention, on peut donner les valeurs que l'on veut aux noms de la colonne à droite (ici préfixées par **sv**_) pour marquer qu'elles se rapportent aux entrées du fichier **/etc/services.** Lors de votre première installation, suivez cette convention ; vous pourrez ensuite en changer selon vos préférences.

/etc/services

Ajouter ces entrées au fichier **/etc/services** de Linux (et non au fichier **$INFORMIXDIR/etc/services**) :

```
sv_rebel_shm 60000/tcp  # Informix IDS 7.30 @ rebel
sv_rebel_net 60001/tcp  # Informix IDS 7.30 @ rebel
```

S'assurer que ces numéros de port peuvent être utilisés ; s'ils ne peuvent l'être, utiliser d'autres numéros de port disponibles. Tout nombre entre 0 et 65536 peut être assigné comme numéro de port, y compris les nombres négatifs. Beaucoup sont préassignés — c'est une sorte de convention dans le monde UNIX — aussi évitez d'utiliser des numéros de port déjà assignés à d'autres canaux Ethernet.

Une fois que vous avez modifié votre fichier **/etc/services,** vous devez redémarrer votre réseau. Dans la plupart des cas, relancer **inetd** le forcera à relire le fichier **/etc/services ;** cependant, un redémarrage peut être nécessaire dans

© copyright Éditions Eyrolles

certaines distributions de Linux. Les diverses distributions ont différents niveaux d'exécution, ce qui signifie que chaque distribution a des procédures uniques de redémarrage du réseau. Une sorte de tradition conduit beaucoup d'administrateurs système Linux à ne pas redémarrer leurs serveurs Linux à froid pour conserver le temps de fonctionnement (*uptime*) : ils s'efforcent de ne redémarrer que si c'est vraiment nécessaire. Dans l'intérêt de la simplicité, et pour réduire les risques d'erreur, redémarrez votre serveur Linux si vous pouvez ; sinon, assurez-vous absolument que la couche réseau de Linux est bien redémarrée, et que le fichier **/etc/services** a été relu. Dans le cas contraire, le serveur Informix ne peut pas parler aux clients sur le réseau, et d'autres problèmes peuvent survenir au cours de l'installation.

Remarquer que les entrées dans le fichier **$INFORMIXDIR/etc/sqlhosts** correspondent aux entrées dans le fichier **/etc/services.** Si elles ne concordent pas, on ne peut pas démarrer le moteur.

/opt/data/rootdbs

Pour utiliser un fichier cooked, il faut en créer un dans le répertoire **/opt/data (**figure 4.6) :

```
$ mkdir /opt/data    # seulement si vous n'avez pas ce répertoire !!!
$ chown informix:informix /opt/data
$ chmod 775 /opt/data
$ cd /opt/data
$ >rootdbs
$ chown informix:informix rootdbs
$ chmod 660 rootdbs
$ ls –l rootdbs
```

Le rootdbs doit maintenant avoir une taille de zéro octet. Informix en est le propriétaire et il est doté de permissions utilisateur et groupe, mais d'aucune permission publique. Il sera **rw-rw----** quand on exécute **ls –l** sur le fichier des rootdbs. Le répertoire, dans lequel résident les rootdbs, sera la propriété d'Informix et l'id du groupe Informix.

Démarrer votre moteur avec oninit

Si vous avez suivi toutes les instructions jusqu'à présent, vous êtes prêt à initialiser votre instance d'Informix. L'initialisation est le processus par lequel la base de données **sysmaster** est créée, et les rootdbs importés en ligne. Il démarrera le moteur et vous pourrez commencer à créer des bases de données et des tables.

© copyright Éditions Eyrolles

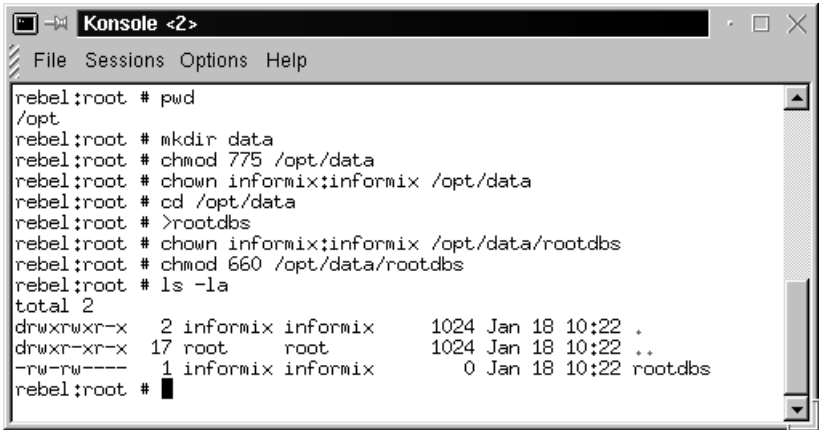
Konsole <2>
File Sessions Options Help
rebel:root # pwd
/opt
rebel:root # mkdir data
rebel:root # chmod 775 /opt/data
rebel:root # chown informix:informix /opt/data
rebel:root # cd /opt/data
rebel:root # >rootdbs
rebel:root # chown informix:informix /opt/data/rootdbs
rebel:root # chmod 660 /opt/data/rootdbs
rebel:root # ls -la
total 2
drwxrwxr-x 2 informix informix 1024 Jan 18 10:22 .
drwxr-xr-x 17 root root 1024 Jan 18 10:22 ..
-rw-rw---- 1 informix informix 0 Jan 18 10:22 rootdbs
rebel:root #

Figure 4–6
Créer le répertoire /opt/data, et créer les rootdbs pour l'initialisation

Vous pouvez maintenant exécuter le programme **oninit** pour initialiser les rootdbs. Souvenez-vous que ce que nous faisons là ne concerne que l'installation initiale. Par la suite, vous n'aurez pas à répéter tous les jours cette initialisation au démarrage. Le programme **oninit** est le même que celui que nous utilisons pour démarrer et arrêter le moteur tous les jours, sans les options d'installation initiale. Il faut être en mode root pour cette opération d'installation ; ensuite, le seul login pour le démarrage quotidien sera **informix**, et **oninit** sera utilisé sans aucune option.

Pour initialiser le moteur, exécuter la commande suivante dans un répertoire quelconque :

```
$ oninit –ivy
```

Pour retenir la combinaison de commande –**ivy**, pensez au poison ivy[1] !

Ne pas s'approcher de l'option –**i** sauf pour exécuter cette initialisation. On ne l'utilise qu'une seule fois — cela pose beaucoup de problème si on l'utilise à nouveau sur la même instance.

- –**i** signifie : initialiser l'instance — ***à ne faire qu'une seule fois !***
- –**v** signifie : mode commentaire ;
- –**y** signifie : répondre oui à toutes les questions.

1. Sumac vénéneux ; le sumac est un arbuste aux nombreuses variétés. *N.d.T.*

© copyright Éditions Eyrolles

Quand on exécute la commande **oninit –ivy,** le dialogue suivant apparaît :

```
rebel:root # oninit –ivy

Checking group membership to determine server run mode succeeded
Reading configuration file '/opt/infor-
mix/ids/etc/rebel_shm.cfg.'..succeeded
Creating /INFORMIXTMP/.infxdirs ... succeeded
Creating infos file "/opt/informix/ids/etc/.infos.rebel_shm" ...
"/opt/informix/ids/etc/.conf.rebel_shm" ... succeeded
Writing to infos file ... succeeded
Checking config parameters...succeeded
Allocating and attaching to shared memory...succeeded
Creating resident pool 398 kbytes...succeeded
Creating buffer pool 402 kbytes...succeeded
Initializing rhead structure...succeeded
Initializing ASF. ..succeeded
Initializing Dictionary Cache and Stored Procedure Cache...succeeded
Bringing up ADM VP...succeeded
Creating VP classes...succeeded
Onlining 0 additional cpu vps...succeeded
Onlining 2 IO vps...succeeded
Forking main_loop thread...succeeded
Initialzing DR structures...succeeded
Forking 1 'ipcshm' listener threads...succeeded
Forking 1 'soctcp' listener threads...succeeded
Starting tracing...succeeded
Initializing 1 flushers...succeeded

Initializing log/checkpoint information...succeeded
Opening primary chunks...succeeded
Opening mirror chunks...succeeded
Initializing dbspaces...succeeded
Validating chunks...succeeded
Creating database partition
Initialize Async Log Flusher...succeeded
Forking btree cleaner...succeeded

rebel:root # Initializing DBSPACETEMP list
Checking database partition index...succeeded
Checking location of physical log...succeeded
Initializing dataskip structure...succeeded

Checking for temporary tables to drop
Forking onmode_mon thread...succeeded
Verbose output complete : mode = 5
```

Immédiatement après ce dernier message, appuyer sur la touche **Entrée** (ou **Return**), et l'invite de commande réapparaît.

© copyright Éditions Eyrolles

Vérifier tout de suite le fonctionnement du moteur avec la commande **onstat.** Comme vous pouvez le voir dans cet exemple, notre moteur fonctionne depuis 26 minutes, 5 secondes :

```
rebel:root # onstat

Informix Dynamic Server Version 7.30.UC10   -- On-Line --
Up 00:26:05 -- 8888 Kbytes

Userthreads
address flags sessid user tty wait tout locks nreads nwrites
10120014 ---P---D 1 root     -        0 0 0 54 2014
101204c8 ---P---F 0 root     -        0 0 0 0 1089
1012097c ---P---- 9 root     -        0 0 0 0 0
10120e30 ---P---B 10 root     -        0 0 0 6 0
10121798 ---P---D 12 root     -        0 0 0 0 0
 5 active, 128 total, 8 maximum concurrent

Profile
dskreads pagreads bufreads %cached  dskwrits pagwrits bufwrits %cached
1764     1765     67300    97.38    1996     4570     19723    89.88

isamtot open   start   read    write   rewrite delete  commit  rollbk
56074   8740   9352    13263   6790    427     1161    772     0

gp_read  gp_write gp_rewrt gp_del   gp_alloc gp_free  gp_curs
0        0        0        0        0        0        0

ovlock   ovuserthread     ovbuff   usercpu  syscpu   numckpts flushes
0        0                71       1.79     0.51     4        17

bufwaits lokwaits lockreqs deadlks  dltouts  ckpwaits compress seqscans
6        0        15805    0        0        1        90       411

ixda-RA  idx-RA   da-RA    RA-pgsused lchwaits
28       0        6        34         1
```

Hourra ! Il fonctionne !

Si le moteur ne fonctionne pas, vous ne verrez pas ce dialogue ! Revenir en arrière et s'assurer que tout est initialisé correctement. Recommencer l'installation si nécessaire et tout vérifier pendant la procédure. Appeler éventuellement le support technique d'Informix pour avoir plus d'aide sur l'installation.

© copyright Éditions Eyrolles

Si vous regardez la taille de **/opt/data/rootdbs**, vous verrez qu'elle est de 20 Mo — ceux que vous avez réservés — alloués dans un gros fichier Linux. Le programme **oninit** a créé le fichier selon vos spécifications.

```
rebel:root # cd /opt/data
rebel:root # ls -l

total 20082
drwxrwxr-x 2 informix informix 1024 Jan 18 10:22 .
drwxr-xr-x 17 root root 1024 Jan 18 10:22 ..
-rw-rw----   1 informix informix 20480000 Jan 18 13:00 rootdbs

rebel:root #
```

Les logs peuvent être vérifiés avec l'option **–l** (L minuscule) :

```
rebel:root # onstat –l

Informix Dynamic Server Version 7.30.UC10   -- On-Line --
Up 00:26:08 -- 8888 Kbytes

Physical Logging
Buffer bufused bufsize numpages numwrits pages/io
  P-1 0 16 212 21 10.10%
      phybegin physize phypos phyused %used
      10003f 500 212 0 0.00

Logical Logging
Buffer bufused bufsize numrecs numpages numwrits recs/pages pages/io
  L-1 0 16 18630 1243 810 15.0 1.5
        Subsystem numrecs Log Space used
        OLDRSAM 18630 1304996

address number flags uniqid begin size used %used
10039324 1 U-B----  1 100233 250 250 100.00%
10039340 2 U-B----  2 10032d 250 250 100.00%
1003935c 3 U-B----  3 100427 250 250 100.00%
10039378 4 U-B----  4 100521 250 250 100.00%
10039394 5 U---C-L  5 10061b 250 243 97.20%
100393b0 6 F------  0 100715 250 0 0.00%
```

Les messages du moteur sont rangés dans le fichier **$INFORMIXDIR/logs/online.log** et peuvent être consultés à l'aide de l'option **onstat –m** :

```
rebel:root # onstat –m

Informix Dynamic Server Version 7.30.UC10   -- On-Line --
Up 00:26:12 -- 8888 Kbytes
```

© copyright Éditions Eyrolles

```
Message Log File: /opt/informix/ids/logs/online.log
Tue Jan 18 11:58:14 2000
11:58:14 Event alarms enabled.
          ALARMPROG = '/opt/informix/ids/etc/log_full.sh'
11:58:19 DR: DRAUTO is 0 (Off)
11:58:19 Requested shared memory segment size rounded from 588KB to 592KB
11:58:19 Informix Dynamic Server Version 7.30.UC10 Software Serial Number
          AAC#A518469
11:58:23 Informix Dynamic Server Initialized—Complete Disk Initialized.
11:58:23 Checkpoint Completed:  duration was 0 seconds.
11:58:23 Dataskip is now OFF for all dbspaces
11:58:23 On-Line Mode
11:58:23 Building 'sysmaster' database ...
11:58:27 Logical Log 1 Complete.
11:58:31 Logical Log 2 Complete.
11:58:31 Checkpoint Completed:  duration was 0 seconds.
11:58:33 Logical Log 3 Complete.
11:58:35 Logical Log 4 Complete.
11:58:41  'sysmaster' database built successfully.
11:58:41  'sysutils' database built successfully.
12:03:52 Checkpoint Completed:  duration was 0 seconds.
12:08:52 Checkpoint Completed :  duration was 0 seconds.
```

À ce stade, le moteur est en route, mais pas totalement prêt à fonctionner dans un premier temps. Il faut encore initialiser certaines options supplémentaires et des paramètres de configuration, tels que les espaces de base de données dont nous avons parlé auparavant dans ce chapitre. Le moteur est déjà utilisable dans le cadre des 20 Mo des rootdbs, mais c'est insuffisant pour une utilisation en tant que serveur de base de données. Parcourir les étapes suivantes, qui rendront le système opérationnel.

Nous avons également parlé du fichier **console.log** ; voici un exemple de sortie qu'il conserve :

```
rebel:root # cat $INFORMIXDIR/logs/console.log

<<Informix Dynamic Server>>> Logical Log 1 Complete.
<<Informix Dynamic Server>>> Logical Log 2 Complete.
<<Informix Dynamic Server>>> Logical Log 3 Complete.
<<Informix Dynamic Server>>> Logical Log 4 Complete.
```

Comme vous pouvez le voir, ces sortes de messages n'ont pas vraiment besoin d'être conservés dans un fichier de log, et la destination dans la ligne **CONSOLE** de votre fichier **ONCONFIG** peut être mise à **/dev/null**. Cela évite d'avoir à gérer la taille de ce fichier.

© copyright Éditions Eyrolles

Pour connaître l'allocation de la mémoire, utiliser **onstat –g seg** :

```
rebel:root # onstat -g seg

Informix Dynamic Server Version 7.30.UC10   -- On-Line --
Up 00:40:09 -- 8888 Kbytes

Segment Summary:
id     key         addr       size      ovhd   class  blkused  blkfree
769    1381386241 10000000   909312    872    R      104      7
770    1381386242 100de000   8192000   724    V      415      585
771    1381386243 108ae000   606208    608    M      67       7
Total:   -        -          9707520   -      -      586      599

   (* segment locked in memory)
```

Cela permet de suivre l'utilisation de la mémoire et son allocation par le serveur IDS. Si le serveur a besoin de plus de mémoire, il allouera l'usage de la mémoire sur la base des options de votre fichier **ONCONFIG**. Nous verrons cela dans la partie suivante de notre installation.

Pour voir l'utilisation du disque par le moteur, utiliser **onstat –d** :

```
rebel:root # onstat –d

Informix Dynamic Server Version 7.30.UC10   -- On-Line --
Up 00:00:20 -- 8888 Kbytes

Dbspaces
address   number flags    fchunk   nchunks  flags    owner    name
1011e13c  1      1        1        1        N        informix rootdbs
 1 active, 2047 maximum

Chunks
address   chk/dbs offset size   free    bpages flags  pathname
1011e1f8  1       1      0      10000   6971   PO-    /opt/data/rootdbs
1 active, 2047 maximum
```

Important. On obtient un listing complet des options **onstat** à l'aide de la commande **onstat –** (sans lettre d'option). Voir le *Guide de l'administrateur* pour un exposé complet de toutes ces options. Ce livre ne traite que des quelques outils qui fournissent le compte rendu le plus complet sur le moteur.

Pour terminer notre configuration, il reste encore quelques dernières modifications à apporter au fichier $**ONCONFIG**, puis nous redémarrerons le moteur.

© copyright Éditions Eyrolles

Arrêter le moteur

Maintenant que le moteur Informix fonctionne, comment peut-on l'arrêter ? Pour cela, utiliser la commande **onmode –ky** :

```
$ onmode -ky
```

Cela permet de faire les changements dans le fichier **$ONCONFIG** avant de redémarrer le moteur.

Toute une série de caractéristiques additionnelles sont proposées dans la commande **onmode**. Voir le *Guide de l'administrateur* pour plus d'informations sur la fonction et l'usage des autres options.

Configuration finale

Que faire ensuite ? Le moteur fonctionne, mais n'est pas encore vraiment opérationnel. La prochaine phase rendra le serveur utilisable, en particulier pour les utilisateurs finaux.

Nous allons initialiser le moteur pour la configuration finale, celle qui devrait convenir à vos besoins à partir d'ici. Assurez-vous que vous avez l'espace disque nécessaire pour les actions qui vont suivre. Si vous avez alloué votre espace disque comme indiqué au début de ce chapitre, vous êtes maintenant prêt à achever le travail.

Rappel

Revoyons rapidement quelques-uns des dbspaces déjà cités.

- **Rootdbs :** créé pendant l'installation et l'initialisation de la base de données.
- **Physdbs :** contiendra le log physique ; doit être transféré hors des rootdbs dans son propre dbspace.
- **Logsdbs :** contiendra les logs logiques ; doit être transféré hors des rootdbs dans leur propre dbspace.
- **Tempdbs :** affecté par défaut aux rootdbs, pointe vers son propre dbspace, et sert pour les tris, les requêtes et les tables temporaires.
- **Dbs_dbs :** contiendra des entrées de base de données, mais aucune entrée de table.

© copyright Éditions Eyrolles

- **Datadbs01 :** contiendra des tables.
- **Indexdbs01 :** contiendra des index.

Physdbs

Pour des questions de performance, on déplace le log physique dans son propre dbspace. Le log physique est un buffer, ou cache, de l'activité de la base de données, c'est-à-dire des données vivantes qui doivent encore être incorporées à la base de données. Le réimplanter dans son propre dbspace permet d'augmenter ou de diminuer sa taille selon les besoins, sans impact sur les rootdbs, et d'augmenter les performances.

Comme nous l'avons vu, les rootdbs contiennent les cerveaux de chaque base de données que vous créez sur un serveur, pour une instance. On peut avoir des instances multiples d'Informix ; chaque instance a ses propres rootdbs et des dbspaces, séparés de ceux des autres instances. C'est important car cela minimise le risque lié aux rootdbs. Réimplanter le log physique est l'une des premières étapes à suivre pour s'assurer que le rootdbs est à son minimum d'activité : garder trace de l'endroit où se trouvent tous les éléments des bases de données et gérer le moteur.

S'assurer que le moteur Informix fonctionne.

```
$ onstat
```

Si le moteur ne fonctionne pas, le démarrer maintenant avec le programme **oninit,** sans *aucune* option ou avec, tout au plus, l'option –**v (**flag commentaire) :

```
$ oninit –v
```

N'utilisez pas l'option –**i** pour exécuter **oninit** après avoir créé les rootdbs : ce serait l'équivalent du reformatage d'un disque physique. Une seule exception : **oninit –v**, qui donne une trace détaillée du démarrage du moteur. Utiliser l'option –**i** sur un serveur de production n'a qu'une utilité : expédier le travail.

Une fois que le moteur fonctionne, créer le premier physdbs avec des commandes UNIX/Linux, et non avec des commandes Informix, comme on l'a fait pour les rootdbs.

© copyright Éditions Eyrolles

Créer un fichier de physdbs

UNIX et Linux vous permettent de créer un fichier de longueur nulle à l'aide de la commande **>nom de fichier**.

```
$ cd /opt/data
$ >physdbs
$ chown informix:informix physdbs
$ chmod 660 physdbs
```

Le fichier est créé, mais il ne contient absolument rien.

La plupart des étapes suivantes se font en mode mono-utilisateur, ou mode *quiescent*. Nous allons nous servir de deux utilitaires — **onspaces** et **onparams** — à utiliser seulement en tant que DBA.

Onspaces peut être utilisé en mode multi-utilisateur, mais **onparams** s'exécute quand le moteur est en mode mono-utilisateur. Pour passer en mode mono-utilisateur, exécuter la commande **onmode –s** — le **s**, ajouté à cette commande, fait disparaître la question suivante :

```
rebel:root # onmode –s

This will perform a GRACEFUL SHUTDOWN -
Do you wish to continue (y/n)? y

rebel:root # onstat

Informix Dynamic Server Version 7.30.UC10  -- Quiescent --
Up 00:22:36 -- 8888 Kbytes

Userthreads
address  flags    sessid user     tty  wait tout locks  nreads  nwrites
10120014 ---P--D  1      informix -    0    0    0      11      5
101204c8 ---P--F  0      informix -    0    0    0      0       2
1012097c ---P---  9      informix -    0    0    0      0       0
10120e30 ---P--B  10     informix -    0    0    0      0       0
10121c4c ---P--D  13     informix -    0    0    0      0       0
 5 active, 128 total, 18 maximum concurrent

Profile
dskreads pagreads bufreads %cached  dskwrits pagwrits bufwrits %cached
20       38       36       44.44    7        7        2        0.00

isamtot open    start   read   write   rewrite delete  commit  rollbk
16      4       5       3      0       0       0       0       0

gp_read  gp_write gp_rewrt gp_del   gp_alloc gp_free  gp_curs
0        0        0        0        0        0        0
```

© copyright Éditions Eyrolles

```
ovlock   ovuserthread      ovbuff   usercpu  syscpu  numckpts flushes
Ø        Ø                 Ø        2.95     Ø.Ø9    1        1Ø

bufwaits lokwaits  lockreqs deadlks  dltouts  ckpwaits compress seqscans
Ø        Ø         1Ø       Ø        Ø        Ø        Ø        Ø

ixda-RA  idx-RA   da-RA    RA-pgsused lchwaits
Ø        Ø        Ø        Ø          Ø
```

Si l'on regarde la première ligne, on voit que Informix IDS est maintenant en mode Quiescent ou mode mono-utilisateur. C'est le seul mode dans lequel on peut se servir de **onparams** pour exécuter certaines commandes parmi les plus critiques, telles que déplacer le log physique dans un autre dbspace. Dans d'autres cas ou sur d'autres plates-formes, **onparams** ne doit être exécuté que si le moteur est totalement arrêté.

Maintenant, créons un physdbs. Pour ce faire, il faut d'abord créer un dbspace 20 Mo pour physdbs avec la commande **onspaces,** puis dérouler le log physique à l'aide de la commande **onparams**. Dans la suite du chapitre, les commandes sont mises en gras dans les exemples de code.

```
rebel:root # onspaces -c -d physdbs -p /opt/data/physdbs -o 0 -s 20480
Verifying physical disk space, please wait ...
Space successfully added.
** WARNING ** A level Ø archive of Root DBSpace will need to be done.

rebel:root # ontape -s -L 0
Archive to tape device '/dev/null' is complete.
Program over.

rebel:root # onstat –l
Informix Dynamic Server Version 7.3Ø.UC1Ø  -- Quiescent --
Up ØØ:46:17 -- 8888 Kbytes

Physical Logging
Buffer   bufused  bufsize  numpages numwrits pages/io
P-1      4        16       4        Ø        Ø.ØØ%
         phybegin physize  phypos   phyused  %used
         1ØØØ3f   5ØØ      223      4        Ø.8Ø

Logical Logging
Buffer   bufused  bufsize  numrecs  numpages numwrits recs/pages pages/io
L-3      Ø        16       21       5        5        4.2        1.Ø
         Subsystem  numrecs  Log Space used
         OLDRSAM    21       1Ø92
```

© copyright Éditions Eyrolles

```
address   number  flags     uniqid  begin    size    used    %used
10039324  1       U-B----   1       100233   250     250     100.00%
10039340  2       U-B----   2       10032d   250     250     100.00%
1003935c  3       U-B----   3       100427   250     250     100.00%
10039378  4       U-B----   4       100521   250     250     100.00%
10039394  5       U-B----   5       10061b   250     250     100.00%
100393b0  6       U---C-L   6       100715   250     7       2.80%
```

Nous avons exécuté une commande **ontape** après la création du dbspace de physdbs. C'est essentiel pour rendre le dbspace de physdbs utilisable. La commande **onstat –l** signale que la sortie du fichier du log logique a aussi changé, en raison de la commande de sauvegarde **ontape**. Exécutons la commande **onstat –d** maintenant, puis après avoir déplacé le log physique, pour pouvoir comparer.

```
rebel:root # onstat -d
Informix Dynamic Server Version 7.30.UC10  -- Quiescent --
Up 00:52:05 -- 8888 Kbytes

Dbspaces
address   number flags    fchunk   nchunks  flags  owner    name
1011e13c  1      1        1        1        N      informix rootdbs
1011efa8  2      1        2        1        N      informix physdbs
 2 active, 2047 maximum

Chunks
address   chk/dbs offset size   free   bpages flags  pathname
1011e1f8  1 1     0      10000  6921          PO-    /opt/data/rootdbs
1011f064  2 2     0      10240  10187         PO-    /opt/data/physdbs
2 active, 2047 maximum
```

Maintenant, on peut voir que l'espace de physdbs existe. Malheureusement, la sortie donne une taille incorrecte ; il suffit de la multiplier par deux pour obtenir les 20 480 Ko que nous avons alloués, soit 20 Mo.

Exécutons **onparams**, sans option, pour consulter les options disponibles :

```
$ onparams
Usage:  onparams  { -a -d DBspace [-s size]        |
                    -d -l logid [-y]               |
                    -p -s size [-d DBspace] [-y] }

    -a  - Add a logical log
    -d  - Drop a logical log
    -p  - Change physical log size and location
    -y  - Automatically responds "yes" to all prompts
```

© copyright Éditions Eyrolles

Nous allons utiliser l'option –**p** pour créer un physdbs de 20 Mo. Souvenez-vous qu'au départ nous avons alloué 50 Mo au log physique, mais que nous n'avons utilisé que 20 Mo, ce qui donne une marge pour l'agrandir.

Pour créer le physdbs, utiliser la commande **onparams** suivante :

```
$ onparams –p –s 50000 –d physdbs –y
```

Voici le résultat de la commande :

```
rebel:root # onparams -p -s 20000 -d physdbs -y
Shutting down, please wait ...
Initializing, please wait ...
Recovering, please wait ...

rebel:root # onstat -d

Informix Dynamic Server Version 7.30.UC10  -- Quiescent --
Up 00:00:31 -- 8888 Kbytes

Dbspaces
address   number  flags    fchunk   nchunks  flags  owner      name
1011e13c  1       1        1        1        N      informix   rootdbs
1011e428  2       1        2        1        N      informix   physdbs
 2 active, 2047 maximum

Chunks
address   chk/dbs offset size   free   bpages flags  pathname
1011e1f8  1 1     0      10000  7421   PO-    /opt/data/rootdbs
1011e348  2 2     0      10240  187    PO-    /opt/data/physdbs
2 active, 2047 maximum
```

Le log physique avait à l'origine une taille de 1 Mo seulement, la taille par défaut installée par le moteur. Maintenant, nous voyons qu'il utilise presque tous les octets du nouveau physdbs de 20 Mo, et que rootdbs a gagné environ 1 Mo d'espace disponible. Quand nous déplacerons les logs logiques, rootdbs gagnera encore plus d'espace. Si l'on regarde dans le répertoire **/opt/data**, on voit le nouveau dbspace du log physique créé : il apparaît comme un autre fichier cooked :

```
rebel:root # ls –l
total 40643
drwxrwxr-x 2 informix informix 1024 Jan 24 11:14 .
drwxr-xr-x 17 root root 1024 Jan 18 10:22 ..
-rw-rw---- 1 informix informix 20971520 Jan 24 12:25 physdbs
-rw-rw---- 1 informix informix 20480000 Jan 24 12:25 rootdbs
```

© copyright Éditions Eyrolles

physdbs est aussi légèrement plus grand physiquement, que **rootdbs** — c'est parce que toutes nos commandes indiquent des tailles en Ko plutôt qu'en Mo. 20 000 Ko est égal à 20 480 000 octets, la taille que nous avons donnée aux rootdbs. 20 480 Ko est égal à 20 971 520 octets, la taille réelle du physdbs que nous avons créé avec la commande **onspaces**, en utilisant une taille en Ko et non en Mo.

À quelques exceptions près, les commandes Informix donnent les informations de taille en Ko et non en Mo. Il faut faire attention à cette caractéristique des commandes Informix et l'utiliser correctement, c'est-à-dire penser d'abord en Ko, pas en Mo. Si vous pensez en Mo, faites-le seulement en termes de conversion des Ko en Mo.

Logsdbs

Créons maintenant un logsdbs comme racine des logs logiques qui résident encore dans les rootdbs. S'assurer que le moteur est en mode *quiescent*, et poursuivre la création.

Nous allons créer 20 nouveaux logs logiques, de 1 Mo chacun, de sorte qu'il nous faudra un dbspace de log logique d'au moins 26 Mo, sinon plus. Il est préférable de le créer un peu plus grand, pour pouvoir ajouter des logs selon les besoins. Notre premier logsdbs sera de 50 Mo, de manière à disposer de réserves pour la croissance. Nous avons six logs logiques dans les rootdbs et nous en voulons 26, nous allons donc en créer 20 nouveaux.

Créons d'abord notre logsdbs :

```
rebel:root # onspaces -c -d logsdbs -p /opt/data/logsdbs -o 0 -s 50000
Verifying physical disk space, please wait ...
Space successfully added.

** WARNING ** A level 0 archive of Root DBSpace will need to be done.
rebel:root # ontape -s -L 0
onstat
Archive to tape device '/dev/null' is complete.

Program over.
rebel:root # onstat –d
Informix Dynamic Server Version 7.30.UC10  -- Quiescent --
Up 02:52:04 -- 8888 Kbytes
```

© copyright Éditions Eyrolles

```
Dbspaces
address  number   flags    fchunk   nchunks  flags owner      name
1011e13c 1        1        1        1        N     informix   rootdbs
1011e428 2        1        2        1        N     informix   physdbs
1011f00c 3        1        3        1        N     informix   logsdbs
 3 active, 2047 maximum

Chunks
address   chk/dbs offset size  free  bpages flags pathname
1011e1f8  1 1     0      10000 7421         PO-   /opt/data/rootdbs
1011e348  2 2     0      10240 187          PO-   /opt/data/physdbs
1011f0c8  3 3     0      25000 24947        PO-   /opt/data/logsdbs
3 active, 2047 maximum
```

Maintenant que le dbspace du logsdbs existe, nous pouvons déplacer les logs logiques, depuis le rootdbs, vers le logsdbs. Quand l'installation a démarré, elle a créé six logs logiques, il reste à en ajouter 20. Pourquoi 26 ? Ajouter 20 logs est un bon exercice, et vous donnera une idée de la manière dont travaille le moteur. Il vaut mieux avoir beaucoup de petits logs logiques ; sur ce point, se reporter au *Guide de l'administrateur.*

Pour déplacer les logs logiques hors des rootdbs, il faut avoir de nouveaux logs en place, déjà disponibles pour le moteur, qui remplaceront les logs logiques existants. Le nombre minimum de logs logiques est de trois. Pour ce faire, nous allons indiquer au moteur quels sont les logs à conserver et ceux à supprimer. Mais nous ne pouvons pas supprimer ou enlever les logs logiques qui sont en cours d'utilisation : il faut donc indiquer au moteur d'utiliser les nouveaux, pour que les anciens puissent être supprimés.

Créer logsdbs

S'assurer que l'on a créé le nouveau dbspace logsdbs avec la commande **onspaces**, comme nous l'avons vu au début de cette section, et que le moteur est en mode mono-utilisateur à l'aide de la commande **onmode –sy :**

```
rebel:root # onmode –sy
```

Exécuter une sauvegarde de niveau zéro pour nettoyer toutes les entrées de log en attente :

```
rebel:root # ontape -s -L 0
Archive to tape device '/dev/null' is complete.

Program over.
```

© copyright Éditions Eyrolles

```
rebel:root # onstat –ld
Informix Dynamic Server Version 7.30.UC10   -- On-Line --
Up 00:19:33 -- 8888 Kbytes

Dbspaces
address   number  flags    fchunk   nchunks  flags  owner      name
1011e13c  1       1        1        1        N      informix   rootdbs
1011e508  2       1        2        1        N      informix   physdbs
1011e5c4  3       1        3        1        N      informix   logsdbs
 3 active, 2047 maximum

Chunks
address   chk/dbs offset size  free   bpages flags pathname
1011e1f8  1 1     0      10000 7421          PO-    /opt/data/rootdbs
1011e348  2 2     0      10240 187           PO-    /opt/data/physdbs
1011e428  3 3     0      25000 24947         PO-   /opt/data/logsdbs
 3 active, 2047 maximum

Physical Logging
Buffer   bufused  bufsize  numpages numwrits  pages/io
P-1      4        16       4        0         0.00%
         phybegin physize  phypos   phyused   %used
         200035   10000    3        4         0.04

Logical Logging
Buffer   bufused  bufsize numrecs numpages numwrits recs/pages pages/io
L-1      0        16      17      3        3        5.7        1.0
         Subsystem  numrecs  Log Space used
         OLDRSAM    17       772

address   number  flags     uniqid  begin    size   used     %used
10039324  1       U-B----   1       100233   250    250      100.00%
10039340  2       U-B----   2       10032d   250    250      100.00%
1003935c  3       U-B----   3       100427   250    250      100.00%
10039378  4       U-B----   4       100521   250    250      100.00%
10039394  5       U-B----   5       10061b   250    250      100.00%
100393b0  6       U---C-L   6       100715   250    17       6.80%
```

Créer des nouveaux logs logiques

Pour créer les logs logiques, répéter la commande **onparams** :

```
rebel:root # onparams -a -d logsdbs -s 1024
Logical log successfully added.

rebel:root # onstat –l
Informix Dynamic Server Version 7.30.UC10   -- Quiescent --
Up 00:23:25 -- 8888 Kbytes
```

© copyright Éditions Eyrolles

```
Physical Logging
Buffer   bufused  bufsize  numpages numwrits  pages/io
P-1      5        16       5        0         0.00%
         phybegin physize  phypos   phyused   %used
         200035   10000    3 5      0.05

Logical Logging
Buffer   bufused bufsize numrecs  numpages numwrits recs/pagespages/io
L-2      0       16      21       4        4        5.2       1.0
         Subsystem  numrecs   Log Space used
         OLDRSAM    21        940

address   number  flags     uniqid  begin    size   used    %used
10039324  1       U-B----   1       100233   250    250     100.00%
10039340  2       U-B----   2       10032d   250    250     100.00%
1003935c  3       U-B----   3       100427   250    250     100.00%
10039378  4       U-B----    4      100521   250    250     100.00%
10039394  5       U-B----   5       10061b   250    250     100.00%
100393b0  6       U---C-L   6       100715   250    18      7.20%
100393cc  7       A------    0      300035   512    0       0.00%
```

À chaque fois que l'on ajoute un log, sans exécuter de sauvegarde de niveau zéro à l'aide de la commande **ontape**, on peut voir le nouveau log logique, en exécutant **onstat –l**, marqué d'un **A**. Ce nouveau log est créé mais n'est pas utilisable tant qu'on n'a pas exécuté de sauvegarde de niveau zéro. Même si la sauvegarde est effectuée dans **/dev/null**, le moteur pourra utiliser le nouveau log logique.

Nous allons créer vingt nouveaux logs logiques de taille 1 Mo, pour un total de 20 Mo dans l'espace des logs logiques. On peut faire cela dans une boucle for, ou répéter vingt fois la commande précédente :

```
for log in 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
do
     LOG_COUNT= `expr $LOG_COUNT + 1 `
     echo $LOG_COUNT
     onparams  -a -d logsdbs -s 1024
done
```

Quelle que soit la manière dont on les ajoute, la commande **onstat** permet de voir les nouveaux logs logiques :

```
rebel:root # onstat –l
Informix Dynamic Server Version 7.30.UC10   -- Quiescent --
Up 00:27:36 -- 8888 Kbytes
```

© copyright Éditions Eyrolles

```
Physical Logging
Buffer   bufused  bufsize numpages  numwrits  pages/io
P-1      1        16      6         1         6.00%
         phybegin physize phypos    phyused   %used
         200035   10000   8         1         0.01

Logical Logging
Buffer   bufused  bufsize numrecs numpages numwrits recs/pages pages/io
L-1      0        16      98      24       24       4.1        1.0
         Subsystem  numrecs  LogSpace used
         OLDRSAM    98       4164

address   number  flags     uniqid  begin    size    used    %used
10039324  1       U-B----   1       100233   250     250     100.00%
10039340  2       U-B----   2       10032d   250     250     100.00%
1003935c  3       U-B----   3       100427   250     250     100.00%
10039378  4       U-B----   4       100521   250     250     100.00%
10039394  5       U-B----   5       10061b   250     250     100.00%
100393b0  6       U---C-L   6       100715   250     38      15.20%
100393cc  7       A------   0       300035   512     0       0.00%
100393e8  8       A------   0       300235   512     0       0.00%
10039404  9       A------   0       300435   512     0       0.00%
10039420  10      A------   0       300635   512     0       0.00%
1003943c  11      A------   0       300835   512     0       0.00%
10039458  12      A------   0       300a35   512     0       0.00%
10039474  13      A------   0       300c35   512     0       0.00%
10039490  14      A------   0       300e35   512     0       0.00%
100394ac  15      A------   0       301035   512     0       0.00%
100394c8  16      A------   0       301235   512     0       0.00%
100394e4  17      A------   0       301435   512     0       0.00%
10039500  18      A------   0       301635   512     0       0.00%
1003951c  19      A------   0       301835   512     0       0.00%
10039538  20      A------   0       301a35   512     0       0.00%
10039554  21      A------   0       301c35   512     0       0.00%
10039570  22      A------   0       301e35   512     0       0.00%
1003958c  23      A------   0       302035   512     0       0.00%
100395a8  24      A------   0       302235   512     0       0.00%
100395c4  25      A------   0       302435   512     0       0.00%
100395e0  26      A------   0       302635   512     0       0.00%
```

rebel:root # onstat -d

```
Informix Dynamic Server Version 7.30.UC10   -- Quiescent --
Up 00:27:41 -- 8888 Kbytes

Dbspaces
address   number  flags    fchunk   nchunks  flags  owner     name
1011e13c  1       1        1        1        N      informix  rootdbs
1011e508  2       1        2        1        N      informix  physdbs
1011e5c4  3       1        3        1        N      informix  logsdbs
 3 active, 2047 maximum
```

© copyright Éditions Eyrolles

```
Chunks
address   chk/dbs offset size   free   bpages flags  pathname
1011e1f8  1 1     0      10000  7421          PO-    /opt/data/rootdbs
1011e348  2 2     0      10240  187           PO-    /opt/data/physdbs
1011e428  3 3     0      25000  14707         PO-    /opt/data/logsdbs
3 active, 2047 maximum
```

On voit que logsdbs est en cours d'utilisation et que rootdbs reste identique. Nous allons maintenant rendre les nouveaux logs disponibles, en exécutant une sauvegarde de niveau zéro. Voici le résultat, avec **onstat –l** :

```
rebel:root # ontape -s -L 0
Archive to tape device '/dev/null' is complete.

Program over.
rebel:root # onstat -l
Informix Dynamic Server Version 7.30.UC10   -- Quiescent --
Up 00:30:20 -- 8888 Kbytes

Physical Logging
Buffer   bufused  bufsize numpages  numwrits  pages/io
P-1      4        16      10        2         5.00%
         phybegin physize phypos    phyused   %used
         200035   10000   9         4         0.04

Logical Logging
Buffer   bufused  bufsize numrecs numpages numwrits recs/pages pages/io
L-1      0        16      115     27       27       4.3       1.0
         Subsystem  numrecs   Log Space used
         OLDRSAM    115       4936

address   number  flags     uniqid  begin    size   used     %used
10039324  1       U-B----   1       100233   250    250      100.00%
10039340  2       U-B----   2       10032d   250    250      100.00%
1003935c  3       U-B----   3       100427   250    250      100.00%
10039378  4       U-B----   4       100521   250    250      100.00%
10039394  5       U-B----   5       10061b   250    250      100.00%
100393b0  6       U---C-L   6       100715   250    41       16.40%
100393cc  7       F------   0       300035   512    0        0.00%
100393e8  8       F------   0       300235   512    0        0.00%
10039404  9       F------   0       300435   512    0        0.00%
10039420  10      F------   0       300635   512    0        0.00%
1003943c  11      F------   0       300835   512    0        0.00%
10039458  12      F------   0       300a35   512    0        0.00%
10039474  13      F------   0       300c35   512    0        0.00%
10039490  14      F------   0       300e35   512    0        0.00%
100394ac  15      F------   0       301035   512    0        0.00%
100394c8  16      F------   0       301235   512    0        0.00%
100394e4  17      F------   0       301435   512    0        0.00%
10039500  18      F------   0       301635   512    0        0.00%
```

© copyright Éditions Eyrolles

```
1003951c  19      F------   0       301835   512     0       0.00%
10039538  20      F------   0       301a35   512     0       0.00%
10039554  21      F------   0       301c35   512     0       0.00%
10039570  22      F------   0       301e35   512     0       0.00%
1003958c  23      F------   0       302035   512     0       0.00%
100395a8  24      F------   0       302235   512     0       0.00%
100395c4  25      F------   0       302435   512     0       0.00%
100395e0  26      F------   0       302635   512     0       0.00%
```

Maintenant, les nouveaux logs logiques peuvent être utilisés, et nous pouvons supprimer ceux des rootdbs. Pour ce faire, il faut indiquer au moteur quels sont les logs à utiliser comme logs courants, à l'aide de la commande **onmode** :

```
rebel:root # onmode

usage:  onmode -abcDdFklMmnOpQRrSsuyZz
      -a <kbytes>    Increase shared memory segment size.
      -b <version>   Revert Dynamic Server disk structures.
      -c [block | unblock]   Do Checkpoint. Block or unblock server.
      -D   <max PDQ priority allowed>
      -d {standard|{primary|secondary <servername>}} set DR server type
      -F Free unused memory segments
      -k Shutdown completely
      -l Force to next logical log
      -M   <decision support memory in kbytes>
      -m Go to multi-user on-line
      -n Set shared memory buffer cache to non-resident
      -O Override space down blocking a checkpoint
      -p <+-#> <class>   Start up or remove virtual processors of
           a specific class
      -Q   <max # decision support queries>
      -R Rebuild the /INFORMIXDIR/etc/.infos.DBSERVERNAME file
      -r Set shared memory buffer cache to resident
      -S   <max # decision support scans>
      -s Shutdown to single user
      -u Shutdown and kill all attached sessions
      -y Do not require confirmation
      -Z <address> heuristically complete specified transaction
      -z <sid>   Kill specified session id
```

Les trois options qui nous intéressent sont **–l**, **–c** et **–m.** Les options **–l** et **–c** gèrent les logs logiques : l'option **–l** force le moteur à utiliser le prochain log logique et l'option **–c** lui indique d'exécuter un checkpoint. Les checkpoints sont décrits dans le *Guide de l'administrateur* ; il s'agit essentiellement d'une marque, que le moteur positionne, sur l'activité du log logique, qui autorise le log à être sauvegardé, et indique comment le récupérer quand on fait une restauration de la base de données. L'option **–m** permet de replacer le serveur en mode multi-utilisateur.

© copyright Éditions Eyrolles

Il ne s'agit pas seulement de transférer l'activité des logs courants vers les nouveaux logs logiques, mais aussi d'exécuter un checkpoint, pour libérer les anciens logs avant de les supprimer. Dès que le moteur n'utilise plus que les nouveaux logs, nous pouvons supprimer les anciens. Il suffit d'exécuter les commandes suivantes sur les anciens logs, jusqu'à ce que **onstat** montre que le moteur se sert des nouveaux logs. Ici, nous devons le faire six fois : la sortie suivante montre les logs, suite à trois commandes **onmode –l** :

```
address   number  flags     uniqid  begin    size    used     %used
10039324  1       U-B----   1       100233   250     250      100.00%
10039340  2       U-B----   2       10032d   250     250      100.00%
1003935c  3       U-B----   3       100427   250     250      100.00%
10039378  4       U-B----   4       100521   250     250      100.00%
10039394  5       U-B----   5       10061b   250     250      100.00%
100393b0  6       U-B---L    6      100715   250     43       17.20%
100393cc  7       U-B----   7       300035   512     1        0.20%
100393e8  8       U-B----   8       300235   512     1        0.20%
10039404  9       U---C--   9       300435   512     0        0.00%
10039420  10      F-----    0       300635   512     0        0.00%
```

On voit dans cet exemple que le log de checkpoint est maintenant le **log 9** dans les nouveaux logs, et que le dernier log sauvegardé est le **log 6**. Exécuter maintenant **onmode –c**, et le dernier log sauvegardé sera le log 9 :

```
address   number  flags     uniqid  begin    size    used     %used
10039324  1       U-B----   1       100233   250     250      100.00%
10039340  2       U-B----   2       10032d   250     250      100.00%
1003935c  3       U-B----   3       100427   250     250      100.00%
10039378  4       U-B----   4       100521   250     250      100.00%
10039394  5       U-B----   5       10061b   250     250      100.00%
100393b0  6       U-B----   6       100715   250     43       17.20%
100393cc  7       U-B----   7       300035   512     1        0.20%
100393e8  8       U-B----   8       300235   512     1        0.20%
10039404  9       U---C-L   9       300435   512     1        0.20%
10039420  10      F-----    0       300635   512     0        0.00%
```

Maintenant que les logs courants sont les nouveaux et non les anciens, on peut supprimer les anciens, comme ceci :

```
$ onparams –d –l 1 –y
$ onparams –d –l 2 –y
$ onparams –d –l 3 –y
$ onparams –d –l 4 –y
$ onparams –d –l 5 –y
$ onparams –d –l 6 –y
```

La commande est répétée pour les logs 1 à 6.

© copyright Éditions Eyrolles

Voici la commande **onparams** sur le dernier log supprimé, le numéro 6, et la commande **onstat –l**, qui montre que ces logs ont disparu :

```
rebel:root # onparams -d -l 6 -y

Logical log 6 successfully dropped.
** WARNING ** A level 0 archive will need to be done of the rootdbs
before level 1 or 2 archive is attempted after dropping the logical log
(see Dynamic Server Administrator's manual)

rebel:root # onstat -l

Informix Dynamic Server Version 7.30.UC10 -- Quiescent --
Up 00:47:29 -- 8888 KbytesPhysical Logging
Buffer   bufused  bufsize numpages  numwrits  pages/io
P-1      0        16      16        9         1.78%
         phybegin physize phypos    phyused   %used
         200035   10000   19        0         0.00

Logical Logging
Buffer   bufused  bufsize numrecs numpages numwrits recs/pages pages/io
L-3      0        16      147     44       44       3.3        1.0
         Subsystem  numrecs   Log Space used
         OLDRSAM    147       6344

address   number  flags     uniqid  begin    size    used    %used
100393cc  7       U-B----   7       300035   512     1       0.20%
100393e8  8       U-B----   8       300235   512     1       0.20%
10039404  9       U---C-L   9       300435   512     13      2.54%
10039420  10      F------   0       300635   512     0       0.00%
1003943c  11      F------   0       300835   512     0       0.00%
10039458  12      F------   0       300a35   512     0       0.00%
10039474  13      F------   0       300c35   512     0       0.00%
10039490  14      F------   0       300e35   512     0       0.00%
100394ac  15      F------   0       301035   512     0       0.00%
100394c8  16      F------   0       301235   512     0       0.00%
100394e4  17      F------   0       301435   512     0       0.00%
10039500  18      F------   0       301635   512     0       0.00%
1003951c  19      F------   0       301835   512     0       0.00%
10039538  20      F------   0       301a35   512     0       0.00%
10039554  21      F------   0       301c35   512     0       0.00%
10039570  22      F------   0       301e35   512     0       0.00%
1003958c  23      F------   0       302035   512     0       0.00%
100395a8  24      F------   0       302235   512     0       0.00%
100395c4  25      F------   0       302435   512     0       0.00%
100395e0  26      F------   0       302635   512     0       0.00%
```

On voit aussi que les rootdbs ont récupéré un peu d'espace, en exécutant la commande **onstat –d** :

```
rebel:root # onstat -d

Informix Dynamic Server Version 7.30.UC10 -- Quiescent --
Up 00:49:54 -- 8888 Kbytes
```

© copyright Éditions Eyrolles

```
Dbspaces
address  number   flags    fchunk   nchunks  flags  owner      name
1011e13c 1        1        1        1        N      informix   rootdbs
1011e508 2        1        2        1        N      informix   physdbs
1011e5c4 3        1        3        1        N      informix   logsdbs
 3 active, 2047 maximum

Chunks
address   chk/dbs offset size  free   bpages flags pathname
1011e1f8  1 1     0      10000 8921          PO-   /opt/data/rootdbs
1011e348  2 2     0      10240 187           PO-    /opt/data/physdbs
1011e428  3 3     0      25000 14707         PO-   /opt/data/logsdbs
3 active, 2047 maximum
```

Maintenant, on doit recréer les logs 1 à 6 dans le nouveau logsdbs, en exécutant six fois cette commande :

```
rebel:root # onparams  -a -d logsdbs -s 1024
```

puis la commande **onstat :**

```
rebel:root # onstat -l
Informix Dynamic Server Version 7.30.UC10 -- Quiescent --
Up 00:52:57 -- 8888 Kbytes

Physical Logging
Buffer   bufused  bufsize numpages  numwrits  pages/io
P-1      1        16      18        10        1.80%
         phybegin physize phypos    phyused   %used
         200035   10000   20        1         0.01

Logical Logging
Buffer   bufused  bufsize numrecs numpages numwrits recs/pages pages/io
L-1      0        16      172     51       51       3.4        1.0
         Subsystem  numrecs   Log Space used
         OLDRSAM    172       7384

address   number  flags     uniqid  begin    size   used     %used
10039324  1       A------   0       302835   512    0        0.00%
10039340  2       A------   0       302a35   512    0        0.00%
1003935c  3       A------   0       302c35   512    0        0.00%
10039378  4       A------   0       302e35   512    0        0.00%
10039394  5       A------   0       303035   512    0        0.00%
100393b0  6       A------   0       303235   512    0        0.00%
100393cc  7       U-B----   7       300035   512    1        0.20%
100393e8  8       U-B----   8       300235   512    1        0.20%
10039404  9       U---C-L   9       300435   512    20       3.91%
10039420  10      F------   0       300635   512    0        0.00%
1003943c  11      F------   0       300835   512    0        0.00%
10039458  12      F------   0       300a35   512    0        0.00%
```

© copyright Éditions Eyrolles

```
10039474  13      F------   0       300c35   512     0        0.00%
10039490  14      F------   0       300e35   512     0        0.00%
100394ac  15      F------   0       301035   512     0        0.00%
100394c8  16      F------   0       301235   512     0        0.00%
100394e4  17      F------   0       301435   512     0        0.00%
10039500  18      F------   0       301635   512     0        0.00%
1003951c  19      F------   0       301835   512     0        0.00%
10039538  20      F------   0       301a35   512     0        0.00%
10039554  21      F------   0       301c35   512     0        0.00%
10039570  22      F------   0       301e35   512     0        0.00%
1003958c  23      F------   0       302035   512     0        0.00%
100395a8  24      F------   0       302235   512     0        0.00%
100395c4  25      F------   0       302435   512     0        0.00%
100395e0  26      F------   0       302635   512     0        0.00%
```

Exécutons une sauvegarde de niveau zéro et c'est fini !

```
rebel:root # ontape -s -L 0
Archive to tape device '/dev/null' is complete.

Program over.
rebel:root # onstat -l
Informix Dynamic Server Version 7.30.UC10   -- Quiescent --
Up 01:03:37 -- 8888 Kbytes

Physical Logging
Buffer   bufused  bufsize numpages  numwrits  pages/io
P-1      4        16      22        11        2.00%
         phybegin physize phypos    phyused   %used
         200035   10000   21        4         0.04

Logical Logging
Buffer   bufused  bufsize numrecs numpages numwrits recs/pages pages/io
L-1      0        16      189     54       54       3.5        1.0
         Subsystem  numrecs    Log Space used
         OLDRSAM    189        8156

address   number  flags     uniqid  begin    size    used     %used
10039324  1       F------   0       302835   512     0        0.00%
10039340  2       F------   0       302a35   512     0        0.00%
1003935c  3       F------   0       302c35   512     0        0.00%
10039378  4       F------   0       302e35   512     0        0.00%
10039394  5       F------   0       303035   512     0        0.00%
100393b0  6       F------   0       303235   512     0        0.00%
100393cc  7       U-B----   7       300035   512     1        0.20%
100393e8  8       U-B----   8       300235   512     1        0.20%
10039404  9       U--C-L-   9       300435   512     23       4.49%
10039420  10      F------   0       300635   512     0        0.00%
1003943c  11      F------   0       300835   512     0        0.00%
10039458  12      F------   0       300a35   512     0        0.00%
```

© copyright Éditions Eyrolles

```
10039474  13      F------   0       300c35   512    0       0.00%
10039490  14      F------   0       300e35   512    0       0.00%
100394ac  15      F------   0       301035   512    0       0.00%
100394c8  16      F------   0       301235   512    0       0.00%
100394e4  17      F------   0       301435   512    0       0.00%
10039500  18      F------   0       301635   512    0       0.00%
1003951c  19      F------   0       301835   512    0       0.00%
10039538  20      F------   0       301a35   512    0       0.00%
10039554  21      F------   0       301c35   512    0       0.00%
10039570  22      F------   0       301e35   512    0       0.00%
1003958c  23      F------   0       302035   512    0       0.00%
100395a8  24      F------   0       302235   512    0       0.00%
100395c4  25      F------   0       302435   512    0       0.00%
100395e0  26      F------   0       302635   512    0       0.00%
```

On peut maintenant consulter l'occupation du disque avec **onstat –d** :

```
rebel:root # onstat -d
Informix Dynamic Server Version 7.30.UC10   -- Quiescent --
Up 01:05:27 -- 8888 Kbytes

Dbspaces
address  number   flags    fchunk   nchunks  flags   owner     name
1011e13c 1        1        1        1        N       informix  rootdbs
1011e508 2        1        2        1        N       informix  physdbs
1011e5c4 3        1        3        1        N       informix  logsdbs
 3 active, 2047 maximum

Chunks
address   chk/dbs offset size   free   bpages flags  pathname
1011e1f8  1 1     0      10000  8921          PO-    /opt/data/rootdbs
1011e348  2 2     0      10240  187           PO-    /opt/data/physdbs
1011e428  3 3     0      25000  11635         PO-    /opt/data/logsdbs
3 active, 2047 maximum
```

On constate qu'un peu d'espace a été gagné sur rootdbs. Il a maintenant près de 17 Mo de libre, ce qui suffit pour gérer le moteur, créer les bases de données **sysmaster** et **sysutils**, utilisées pour gérer les sauvegardes Informix.

Tempdbs

Les moteurs Informix doivent disposer d'un espace de travail pour le tri des requêtes et d'un endroit où placer les tables temporaires. Si l'on ne précise rien pour ce faire, le moteur utilisera **/tmp** et son propre espace, appelé **INFORMIXTEMP**, situé habituellement dans le répertoire **/.** Mais ce n'est pas correct : il faut créer un tempdbs qui permette au système de contrôler où se produiront les tris et où seront placées les tables temporaires.

© copyright Éditions Eyrolles

La création de ce dbspace est un peu différente de ce qui précède. Il faut indiquer au moteur d'utiliser un dbspace provisoire ; sinon il le traitera comme un dbspace normal. Pour cela on ajoute l'option –**t** à la commande **onspaces** :

```
$ onspaces -c -t -d tempdbs    -p /opt/data/tempdbs  -o 0 -s 102400
```

En voici le résultat :

```
rebel:root # onspaces -c -t -d tempdbs -p /opt/data/tempdbs -o 0 -s 300000
Verifying physical disk space, please wait ...
Space successfully added.
rebel:root # ls -l
total 392018
drwxrwxr-x 2 informix informix 1024 Jan 24 17:41 .
drwxr-xr-x 17 root root 1024 Jan 24 17:41 ..
-rw-rw----   1 informix informix 51200000 Jan 24 17:52 logsdbs
-rw-rw----   1 informix informix 20971520 Jan 24 17:52 physdbs
-rw-rw----   1 informix informix 20480000 Jan 24 17:52 rootdbs
-rw-rw----   1 informix informix 307200000 Jan 24 17:52 tempdbs
```

tempdbs est maintenant créé : il sera utilisable après quelques dernières modifications du fichier **$INFORMIXDIR/etc/$ONCONFIG.**

Nous répéterons ces étapes pour l'espace **dbs_dbs** et le dbspace **datadbs01,** et la création de dbspace sera terminée. Il restera encore à faire les corrections de **$INFORMIXDIR/etc/$ONCONFIG**, et l'on pourra commencer à créer des bases de données !

Mettons ce qui suit dans un script :

```
onspaces -c -d dbs_dbs    -p /opt/data/dbs_dbs    -o 0 -s 100000
onspaces -c -d datadbs01  -p /opt/data/datadbs01  -o 0 -s 250000
onspaces -c -d indexdbs01 -p /opt/data/indexdbs01 –o 0 -s 250000
ontape –s –L 0
```

et voici la trace de son exécution :

```
onspaces -c -d dbs_dbs -p /opt/data/dbs_dbs -o 0 -s 100000
Verifying physical disk space, please wait ...
Space successfully added.

** WARNING ** A level 0 archive of Root DBSpace will need to be done.
onspaces -c -d datadbs01 -p /opt/data/datadbs01 -o 0 -s 250000
Verifying physical disk space, please wait ...
Space successfully added.

** WARNING ** A level 0 archive of Root DBSpace will need to be done.
```

© copyright Éditions Eyrolles

```
onspaces -c -d indexdbs01 -p /opt/data/indexdbs01 -o 0 -s 250000
Verifying physical disk space, please wait ...
Space successfully added.

** WARNING ** A level 0 archive of Root DBSpace will need to be done.
rebel:root # ontape -s -L 0
Archive to tape device '/dev/null' is complete.

Program over.
```

Exécutons à nouveau la commande **onstat** :

```
rebel:root # onstat -d
Informix Dynamic Server Version 7.30.UC10   -- Quiescent --
Up 01:36:42 -- 8888 Kbytes

Dbspaces
address  number   flags    fchunk   nchunks  flags  owner      name
1011e13c 1        1        1        1        N      informix   rootdbs
1011e508 2        1        2        1        N      informix   physdbs
1011e5c4 3        1        3        1        N      informix   logsdbs
1039738c 4        2001     4        1        N T    informix   tempdbs
103977ec 5        1        5        1        N      informix   dbs_dbs
10397ca8 6        1        6        1        N      informix   datadbs01
1039b21c 7        1        7        1        N      informix   indexdbs01
 7 active, 2047 maximum

Chunks
address   chk/dbs offset size    free bpages flags  pathname
1011e1f8  1 1     0      10000   8921        PO-    /opt/data/rootdbs
1011e348  2 2     0      10240   187         PO-    /opt/data/physdbs
1011e428  3 3     0      25000   11635       PO-    /opt/data/logsdbs
10397448  4 4     0      150000  149947      PO-    /opt/data/tempdbs
103978a8  5 5     0      50000   49947       PO-    /opt/data/dbs_dbs
10397d64  6 6     0      125000  124947      PO-    /opt/data/datadbs01
1039b2d8  7 7     0      125000  124947      PO-    /opt/data/indexdbs01
7 active, 2047 maximum
```

Tous les espaces de bases de données essentiels sont maintenant créés et utilisables. Il ne reste plus qu'un petit problème : terminer la mise à jour de quelques options du fichier **$INFORMIXDIR/etc/$ONCONFIG**, et le système sera prêt. Par exemple, l'espace tempdbs n'est pas utilisé tant qu'il n'est pas activé dans **$ONCONFIG**.

Arrêtons le moteur par la commande **onmode –ky :**

```
$ onmode –ky
```

© copyright Éditions Eyrolles

Dernières options de configuration de $ONCONFIG

Editez le fichier **$INFORMIXDIR/etc/$ONCONFIG** et effectuez les modifications suivantes.

Log physique

Remarquons que les options de physdbs ont été mises à jour :

```
# Physical Log Configuration

PHYSDBS physdbs          # Location (dbspace) of physical log
PHYSFILE 20000           # Physical log file size (Kbytes)
```

Log logique

Le nombre de **LOGFILES** a été mis à jour, mais **LOGSIZE** n'a pas changé : il vaut mieux lui donner la valeur qu'il aura dans le système.

- Valeurs par défaut :

```
# Logical Log Configuration

LOGFILES 26              # Number of logical log files
LOGSIZE 500              # Logical log size (Kbytes)
```

- Nouvelles valeurs :

```
# Logical Log Configuration

LOGFILES 26              # Number of logical log files
LOGSIZE 1024             # Logical log size (Kbytes)
```

Dans notre cas, indiquons 1024, c'est-à-dire une taille de 1 Mo. Voir le *Guide de l'administrateur* pour les détails sur la taille des logs logiques.

Utilisation de la mémoire

La taille par défaut de 8 Mo est très minimaliste : il vaut mieux l'augmenter — essayez différentes valeurs pour trouver celle qui convient le mieux :

- taille par défaut :

```
SHMVIRTSIZE 8000         # initial virtual shared memory segment size
```

- nouvelle taille :

```
SHMVIRTSIZE 24000        # initial virtual shared memory segment size
```

© copyright Éditions Eyrolles

On peut laisser la valeur par défaut et tester d'abord toutes les autres options, puis agrandir ensuite la taille de la mémoire. Dans les exemples présentés à la fin de cette section, la trace du redémarrage du moteur montre que la valeur a maintenant changé. Si cela vous pose un problème, ne changez ce paramètre qu'à la fin.

Options LRU

Se reporter au *Guide de l'administrateur* et aux autres documentations Informix au sujet des concepts sous-jacents à ces nouvelles options. Nous rappelons que l'accent est mis ici sur l'installation, et que seules sont données les quelques options nécessaires à des performances initiales correctes — il vous appartient d'approfondir le reste.

- Valeurs par défaut :

```
LRUS 8                  # Number of LRU queues
LRU_MAX_DIRTY 60        # LRU percent dirty begin cleaning limit
LRU_MIN_DIRTY 50        # LRU percent dirty end cleaning limit
```

  Ces options sont traitées dans plusieurs ouvrages dédiés à Informix qui vous aideront énormément. On trouve aussi une discussion intéressante dans le groupe de news USENET comp.databases.informix, à propos de l'origine de ces options. Les valeurs données ici reposent sur l'expérience et sur le constat d'une amélioration des performances.

- Nouvelles valeurs :

```
LRUS 4                  # Number of LRU queues
LRU_MAX_DIRTY 5         # LRU percent dirty begin cleaning limit
LRU_MIN_DIRTY 1         # LRU percent dirty end cleaning limit
```

Options DBSPACETEMP

Il faut modifier cette option de manière à ce que le nouveau dbspace tempdbs soit utilisé pour les tables temporaires et le tri des requêtes.

- Valeur par défaut :

```
# DBSPACETEMP                 # Default temp dbspaces
```

- Nouvelle valeur :

```
DBSPACETEMP tempdbs
```

© copyright Éditions Eyrolles

Redémarrer le moteur

On redémarre le moteur Informix par la commande **oninit –v** :

```
rebel:root # oninit –v
Checking group membership to determine server run modesucceeded
Reading configuration file '/opt/infor-
mix/ids/etc/rebel_shm.cfg.'..succeeded
Creating /INFORMIXTMP/.infxdirs ... succeeded
Creating infos file "/opt/informix/ids/etc/.infos.rebel_shm" ...
"/opt/informix/ids/etc/.conf.rebel_shm" ... succeeded
Writing to infos file ... succeeded
Checking config parameters...succeeded
Allocating and attaching to shared memory...succeeded
Creating resident pool 398 kbytes...succeeded
Creating buffer pool 402 kbytes...succeeded
Initializing rhead structure...succeeded
Initializing ASF. ..succeeded
Initializing Dictionary Cache and Stored Procedure Cache...succeeded
Bringing up ADM VP...succeeded
Creating VP classes...succeeded
Onlining 0 additional cpu vps...succeeded
Onlining 2 IO vps...succeeded
Forking main_loop thread...succeeded
Initialzing DR structures...succeeded
Forking 1 'ipcshm' listener threads...succeeded
Forking 1 'soctcp' listener threads...succeeded
Starting tracing...succeeded
Initializing 1 flushers...succeeded
Initializing log/checkpoint information...succeeded
Opening primary chunks...succeeded
Opening mirror chunks...succeeded
Initializing dbspaces...succeeded
Validating chunks...succeeded
Initialize Async Log Flusher...succeeded
Forking btree cleaner...succeeded
Initializing DBSPACETEMP list
rebel:root # Checking database partition index...succeeded
Checking location of physical log...succeeded
Initializing dataskip structure...succeeded
Checking for temporary tables to drop
Forking onmode_mon thread...succeeded
Verbose output complete: mode = 5
```

On peut ensuite voir les changements avec la commande **onstat –m** :

```
rebel:root # onstat –m
Informix Dynamic Server Version 7.30.UC10   -- On-Line --
Up 00:01:11 -- 8888 Kbytes
```

© copyright Éditions Eyrolles

```
Message Log File: /opt/informix/ids/logs/online.log

Mon Jan 24 18:25:43 2000

18:25:43 Event alarms enabled.
         ALARMPROG = '/opt/informix/ids/etc/log_full.sh'
18:25:48 DR: DRAUTO is 0 (Off)
18:25:48 Requested shared memory segment size rounded from 588KB to
592KB
18:25:49 Informix Dynamic Server Version 7.30.UC10 Software Serial Number
         AAC#A518469
18:25:50 Informix Dynamic Server Initialized -- Shared Memory Initialized.
18:25:50 Physical Recovery Started.
18:25:50 Physical Recovery Complete: 0 Pages Restored.
18:25:50 Logical Recovery Started.
18:25:53 Logical Recovery Complete.
       0 Committed, 0 Rolled Back, 0 Open, 0 Bad Locks

18:25:53 Onconfig parameter LOGFILES modified from 6 to 26.
18:25:53 Onconfig parameter LOGSIZE modified from 500 to 1024.
18:25:53 Onconfig parameter DBSPACETEMP modified from <blank> to tempdbs.
18:25:53 Dataskip is now OFF for all dbspaces
18:25:53 On-Line Mode
18:25:53 Checkpoint Completed :  duration was 0 seconds.
```

Tout fonctionne bien, sauf que je veux augmenter l'utilisation de la mémoire : je modifie donc le fichier **$ONCONFIG** et je redémarre le moteur. La sortie de **onstat –m** montre le changement, ainsi que la première ligne du compte rendu de **onstat** :

```
rebel:root # onmode –ky
rebel:root # oninit –v
rebel:root # onstat –m

Informix Dynamic Server Version 7.30.UC10   -- On-Line --
Up 00:00:14 -- 24888 Kbytes

Message Log File: /opt/informix/ids/logs/online.log
18:25:53 Checkpoint Completed:  duration was 0 seconds.
18:27:12 Informix Dynamic Server Stopped.

Mon Jan 24 18:27:33 2000

18:27:33 Event alarms enabled.
         ALARMPROG = '/opt/informix/ids/etc/log_full.sh'
18:27:39 DR: DRAUTO is 0 (Off)
18:27:39 Requested shared memory segment size rounded from 588KB to 592KB
18:27:39 Informix Dynamic Server Version 7.30.UC10 Software Serial Number
         AAC#A518469
```

© copyright Éditions Eyrolles

```
18:27:40 Informix Dynamic Server Initialized -- Shared Memory Initialized.
18:27:41 Physical Recovery Started.
18:27:41 Physical Recovery Complete: 0 Pages Restored.
18:27:41 Logical Recovery Started.
18:27:44 Logical Recovery Complete.
        0 Committed, 0 Rolled Back, 0 Open, 0 Bad Locks

18:27:44 Onconfig parameter SHMVIRTSIZE modified from 8000 to 24000.
18:27:44 Dataskip is now OFF for all dbspaces
18:27:44 On-Line Mode
18:27:44 Checkpoint Completed:  duration was 0 seconds.
```

rebel:root # onstat -g seg

```
Informix Dynamic Server Version 7.30.UC10   -- On-Line --
Up 00:09:24 -- 24888 Kbytes

Segment Summary:
id     key         addr     size      ovhd  class  blkused  blkfree
1537   1381386241  10000000 909312    872   R      104      7
1538   1381386242  100de000 24576000  972   V      504      2496
1539   1381386243  1184e000 606208    608   M      67       7
Total: -           -        26091520   -      -      675     2510

   (* segment locked in memory)
```

Votre système est maintenant opérationnel et il peut être mis entre les mains de vos utilisateurs finaux !

Voici quelques derniers conseils. Quand on crée une base de données, il faut la créer dans le dbspace **dbs_dbs** ; une table sera créée dans **datadbs01** ; un index, dans **indexdbs01**. Ces opérations sont décrites dans la documentation Informix : pour ce faire, on se sert du SQL standard et des commandes SQL spécifiques des moteurs Informix. Les options d'initialisation qui ont été retenues ici donnent la meilleure souplesse et laissent des réserves d'espace pour la croissance du système.

Autres outils

Voici une liste de quelques outils fournis avec le moteur de base de données Informix.

- **onlog** : explorateur de log logique qui permet de voir le contenu des logs logiques. Il suffit d'exécuter la commande **onlog** pour afficher le contenu des logs logiques.

© copyright Éditions Eyrolles

- **dbschema** : générateur de rapport destiné au DBA ; il produit des instructions SQL, utilisées pour créer des bases de données et des tables.
- **dbexport, dbimport, dbload** : ces outils permettent de sauvegarder en totalité la base de données dans un fichier aisément utilisable pour restaurer la base de données avec un minimum de travail.
- **finderr** : ce programme produit une documentation détaillée des messages d'erreurs d'Informix. On exécute **finderr** en donnant un numéro d'erreur et il produit une petite page d'information sur la nature du problème.
- **tables SMI** : vous allez vite vous familiariser avec la base de données sysmaster située dans le dbspace des rootdbs. Elle contient en abondance des informations sur le moteur. Par exemple, on peut obtenir un compte rendu de l'utilisation du disque un peu plus lisible.

Le rapport suivant, appelé **IDSspaces**, donne un compte rendu agréable de l'utilisation du disque. Au lieu de ce rapport :

```
rebel:root # onstat -d

Informix Dynamic Server Version 7.30.UC10   -- On-Line --
Up 00:38:51 -- 24888 Kbytes

Dbspaces
address  number   flags    fchunk   nchunks  flags  owner      name
1011e13c 1        1        1        1        N      informix   rootdbs
1011e888 2        1        2        1        N      informix   physdbs
1011e944 3        1        3        1        N      informix   logsdbs
1011ea00 4        2001     4        1        N T    informix   tempdbs
1011eabc 5        1        5        1        N      informix   dbs_dbs
1011eb78 6        1        6        1        N      informix   datadbs01
1011ec34 7        1        7        1        N      informix   indexdbs01
 7 active, 2047 maximum

Chunks
address   chk/dbs offset size   free   bpages flags  pathname
1011e1f8  1   1   0      10000  8921          PO-    /opt/data/rootdbs
1011e348  2   2   0      10240  187           PO-    /opt/data/physdbs
1011e428  3   3   0      25000  11635         PO-    /opt/data/logsdbs
1011e508  4   4   0      150000 149947        PO-    /opt/data/tempdbs
1011e5e8  5   5   0      50000  49947         PO-    /opt/data/dbs_dbs
1011e6c8  6   6   0      125000 124947        PO-    /opt/data/datadbs01
1011e7a8  7   7   0      125000 124947        PO-    /opt/data/indexdbs01
7 active, 2047 maximum
```

© copyright Éditions Eyrolles

on obtient ceci :

```
rebel:root # IDSspaces
Informix DB Space Report Mon Jan 24 18:39:14 EST 2000
datadbs01
DB  Chk DB          -------- Pages -------- Pct   ------ Kilobytes -----
No  No  Space Name  Total    Free    Used   Full   Total   Free    Used
6   6   datadbs01   125000   124947  53     0.04   256000  255891  108
         datadbs01  125000   124947  53     0.04   256000  255891  108

dbs_dbs
DB  Chk DB          -------- Pages -------- Pct   ----- Kilobytes ------
No  No  Space Name  Total    Free    Used   Full   Total   Free    Used
5   5   dbs_dbs     50000    49947   53     0.11   102400  102291  108
         dbs_dbs    50000    49947   53     0.11   102400  102291  108

indexdbs01
DB  Chk DB          -------- Pages -------- Pct ------- Kilobytes ------
No  No  Space Name  Total    Free    Used   Full   Total   Free    Used
7   7   indexdbs01  125000   124947  53     0.04   256000  255891  108
         indexdbs01 125000   124947  53     0.04   256000  255891  108

logsdbs
DB  Chk DB          -------- Pages -------- Pct  ------ Kilobytes ------
No  No  Space Name  Total    Free    Used   Full   Total   Free    Used
3   3   logsdbs     25000    11635   13365  53.46  51200   23828   27371
         logsdbs    25000    11635   13365  53.46  51200   23828   27371

physdbs
DB  Chk DB          -------- Pages -------- Pct  ------ Kilobytes ------
No  No  Space Name  Total    Free    Used   Full   Total   Free    Used
2   2   physdbs     10240    187     10053  98.17  20971   382     20588
         physdbs    10240    187     10053  98.17  20971   382     20588

rootdbs
DB  Chk DB          -------- Pages -------- Pct  ------ Kilobytes ------
No  No  Space Name  Total    Free    Used   Full   Total   Free    Used
1   1   rootdbs     10000    8921    1079   10.79  20480   18270   2209
         rootdbs    10000    8921    1079   10.79  20480   18270   2209

tempdbs
DB  Chk DB          -------- Pages -------- Pct  ------ Kilobytes ------
No  No  Space Name  Total    Free    Used   Full   Total   Free    Used
4   4   tempdbs     150000   149939  61     0.04   307200  307075  124
         tempdbs    150000   149939  61     0.04   307200  307075  124
```

Voici une autre vue qui utilise un programme appelé **IDSchunks** :

```
rebel:informix $ IDSchunks 3
Linux rebel 2.2.13 #1 Mon Nov 8 15:51:29 CET 1999 i686 unknown
/opt/informix/ids/etc/rebel_shm.cfg
Report 3: Order By Data Space Name, Database, Chunk, File
```

© copyright Éditions Eyrolles

```
datadbs01
  Dbs  Chk  Filesystem              Total Pages  Free Pages  % Free
    6  6    /opt/data/datadbs01     125000       124947      %  99.96

datadbs02
  Dbs  Chk  Filesystem              Total Pages  Free Pages  % Free
    8  8    /opt/data/datadbs02     25000        24947       %  99.79
    8  9    /opt/data/datadbs02.02  25000        24997       %  99.99
    8  10   /opt/data/datadbs02.03  25000        24997       %  99.99
                                    75000        74941       %  99.92

dbs_dbs
  Dbs  Chk  Filesystem              Total Pages  Free Pages  % Free
    5  5    /opt/data/dbs_dbs       50000        49947       %  99.89

indexdbs01
  Dbs  Chk  Filesystem              Total Pages  Free Pages  % Free
    7  7    /opt/data/indexdbs01    125000       124947      %  99.96

logsdbs
  Dbs  Chk  Filesystem              Total Pages  Free Pages  % Free
    3  3    /opt/data/logsdbs       25000        11635       %  46.54

physdbs
  Dbs  Chk  Filesystem              Total Pages  Free Pages  % Free
    2  2    /opt/data/physdbs       10240        187         %  1.83

rootdbs
  Dbs  Chk  Filesystem              Total Pages  Free Pages  % Free
    1  1    /opt/data/rootdbs       10000        8577        %  85.77

tempdbs
  Dbs  Chk  Filesystem              Total Pages  Free Pages  % Free
    4  4    /opt/data/tempdbs       150000       149947      %  99.96

            Total All Chunks        570240       545128      %  95.60
```

Remarquons, dans ce compte rendu, que deux chunks ont été ajoutés au dbspace **datadbs02**, pour illustrer la sortie. Ce compte rendu et d'autres outils d'administration sont disponibles à http://www.inxutil.com. **IDSspaces** est un script de shell, qui formule une requête au moteur de base de données et formate l'information dans un format plus lisible que celui produit par onstat. Si vous téléchargez **IDSspaces**, vous aurez, en plus, le code source !

Dbaccess

L'un des outils les plus intéressants fournis avec le moteur est **Dbaccess**. C'est une application, en mode caractère, qui permet de faire des prélèvements directs sur le moteur, une base de données ou des tables. On peut aussi faire des requêtes sur les tables et des opérations de maintenance de base de

© copyright Éditions Eyrolles

données sur toutes les bases de données d'une instance particulière. La documentation de Dbaccess est disponible dans le *Guide d'utilisation de Dbaccess*. Quand on exécute **Dbaccess**, on obtient l'affichage suivant :

```
DBACCESS  :   Query-language  Connection Database Table Session Exit
Use SQL query language.

---------------------- Press CTRL-W for Help. ---
```

Dbaccess est facile à utiliser et fonctionne bien dans une fenêtre VT-100. Il vous offre un excellent bloc-notes pour travailler sur votre base de données. Il existe aussi des outils graphiques, en cours de développement chez Informix, qui seront peut-être inclus dans la distribution du moteur.

Onmonitor

Autre outil que nous n'avons pas utilisé au cours de l'installation : **onmonitor**. Il permet de surveiller le moteur et d'effectuer des opérations de maintenance, mais il n'est ni très souple ni très facile à utiliser.

Dans **onmonitor**, on peut voir ou changer les nombreuses options du moteur ; cependant, la plupart des administrateurs Informix optent pour le contrôle en ligne de commande. Dans un environnement de production, plus vous avez de contrôles, plus vous prenez de risques, tout simplement parce qu'une seule mauvaise commande peut effacer toutes les données de la société. Exécuter **onmonitor** : la sortie suivante apparaîtra dans une interface en mode caractère, sous la même forme que les lignes de Dbaccess :

```
Dynamic Server :   Status Parameters Dbspaces Mode Force-Ckpt  ...
Status menu to view Dynamic Server.

-------------------------On-Line------- Press CTRL-W for Help. ---
```

Par exemple, **onmonitor** donne les mêmes informations sur la base que la commande **onstat** :

```
STATUS :   Profile Userthreads Spaces Databases Logs Archive  ...
Display system profile information.

-------------------------On-Line------- Press CTRL-W for Help. ---
      Boot Time Mon Jan 24 18:27:33 2000
   Current Time Mon Jan 24 18:56:48 2000

Disk Reads  Buff.Reads   %Cached   Disk Writes Buff. Writes  %Cached
108         1167         90.75     22          22            0.00
```

© copyright Éditions Eyrolles

```
Over Lock   Over Userthread  User Time   Sys. Time    Checkpoints
Ø           Ø                3           Ø            3

ixda-RA     idx-RA     da-RA      RA-pgsused   Latch Waits
5           Ø          Ø          5            Ø

Buff. Waits Lock Waits   Lock Req.   Deadlocks  DLTimeouts Check Waits
2           Ø            356         Ø          Ø          Ø

CALLS:   Open     Start    Read     Write
         168      211      354      7
                                                Total Calls
         Rewrite  Delete   Commit   Rollback    1Ø51
         Ø        Ø        Ø        Ø
```

C'est un programme dont vous aurez rarement besoin si vous apprenez à utiliser la commande **onstat** et les options de **$ONCONFIG.** Il est beaucoup plus facile et beaucoup moins risqué d'éditer le fichier **$ONCONFIG** à la main et de redémarrer le moteur. Il y a certaines choses que l'on peut faire avec la commande **onmode** pour éviter de redémarrer le moteur, sans pour autant utiliser **onmonitor**. Il est important de revoir toutes les commandes et les programmes disponibles dans le répertoire **$INFORMIXDIR/bin.** Un outil web d'administration du système est maintenant disponible pour ceux qui veulent gérer leur serveur avec un navigateur web.

Ressources

Quand on a besoin d'aide sur les produits Informix, on dispose de plusieurs sortes de ressources : les unes se trouvent auprès de l'assistance client, les autres sont les contacts avec d'autres clients ou professionnels d'Informix.

Support technique Informix

Vous pouvez appeler le support technique d'Informix au 1-800-331-1763 pour obtenir l'assistance directe d'un ingénieur du support client sur un problème particulier. Le support technique d'Informix a gagné des prix ces dernières années, pour son temps de réponse exceptionnel et les solutions apportées.

© copyright Éditions Eyrolles

Site web Informix

Une visite sur le site web d'Informix à http://www.informix.com vous permettra de tirer profit d'autres méthodes de l'assistance client. Il y a sur ce site un réseau de développeurs Informix qui couvre de nombreux sujets, et souvent, un groupe de discussion à l'accès limité pour chacun des domaines du produit. Il existe aussi un centre TechInfo, un bon moyen d'obtenir des informations sur les corrections de bogues et autres informations techniques.

Groupe de news Usenet comp.databases.informix

C'est un groupe de news accessible sur l'Internet *via* un serveur de news, une liste de diffusion e-mail ou un serveur de news web, tel DejaNews. On peut en savoir plus sur ce groupe de news en allant visiter le site web IIUG (*International Informix Users Group*) du groupe international des utilisateurs Informix à http://www.iiug.org, ou en visitant un serveur de news du web.

Groupe international des utilisateurs Informix (IIUG)

L'IIUG à http://www.iiug.org est une mine de renseignements très fréquentée par ceux qui se servent des produits Informix. Non seulement on y trouve plusieurs listes e-mail pour souscrire un abonnement, mais aussi une abondance d'outils pour développer son propre monde Informix, des liens vers les groupes d'utilisateurs locaux — l'un d'entre eux est probablement proche de vous — et des recommandations pour démarrer son propre groupe d'utilisateurs.

Presse Informix

Plusieurs livres sont à la disposition des utilisateurs Informix. La plupart d'entre eux sont écrits par les meilleurs professionnels du monde Informix, qui ont des années d'expérience des produits Informix. On trouve sur le site web Informix les nouveaux titres ainsi que les parutions disponibles dans la plupart des grandes librairies et des librairies web. Informix produit aussi la plus grande part de sa documentation sur des ensembles de CD-ROM que l'on peut commander chez Informix, pour un prix moins élevé que les documents imprimés. Vous pouvez contacter votre bureau de ventes le plus proche pour commander les CD-ROM.

© copyright Éditions Eyrolles

Formation Informix

Informix propose l'une des meilleures formations dans son domaine ; elle a même été primée. Des gens disposant de peu d'expérience sur les produits Informix ont été formés très rapidement et parfaitement à l'utilisation des produits, en suivant les cours proposés par Informix. Certains cours sur les bases de données sont assez génériques pour s'appliquer à d'autres produits, et offrent une base très sérieuse, sur laquelle bâtir une connaissance solide en matière de base de données. Si votre préoccupation essentielle concerne les performances, ou si vous voulez approfondir vos connaissances du moteur de base de données, prenez la formation directement applicable dans le cadre de votre travail. Informix offre aussi des programmes de certification pour assurer que votre effort et vos compétences sont reconnues dans un cadre tel que celui des directions qui souhaitent apprécier votre professionnalisme. Plus vous en savez, plus vous serez reconnu.

Résumé

Maintenant que vous avez installé IDS pour Linux, que pouvez-vous en faire ? Eh bien, pour commencer, connecter au serveur n'importe quelle interface utilisateur compatible ODBC et commencer immédiatement à développer ou à connecter des applications au moteur de base de données ; créer et connecter des applications web et des clients directement au moteur avec ODBC, JDBC ou des connexions réseau I-connect. Le domaine des applications n'est limité que par votre imagination.

Informix IDS 7.3x est un moteur de base de données SQL très puissant, avec tout ce dont a besoin un serveur de base de données professionnel. Ce serveur présente des qualités de haute fiabilité et de robustesse adaptées à une grande variété d'opérations de stockage et de recherche de données, impliquant des taux d'accès de bas ou de haut volume. Certains des serveurs de base de données les plus rapides en usage sont des serveurs Informix sur UNIX. Informix a dépassé les serveurs concurrents d'autres éditeurs et continue à mener le bal. La version Linux d'IDS n'est pas différente, en interne, de celles des autres plates-formes UNIX ; les seules limites auxquelles on doit faire face sont le noyau Linux courant et la plate-forme matérielle.

IDS 7.3 peut traiter de un à N utilisateurs, la seule limite étant, bien sûr, le matériel. On peut tout apprendre de la version UNIX avec Linux, à l'exception du chargeur haute performance et des disques bruts, qui ne sont pas encore disponibles sur Linux. IDS est compatible à un certain degré avec le moteur standard (SE, *Standard Engine*), mais seulement pour les caractéristi-

© copyright Éditions Eyrolles

ques de base. Le moteur standard est plus limité et propose moins de types de données que le moteur à mémoire partagée de IDS. Le moteur standard est passif, à base d'ISAM. SE est très fiable mais ne dispose pas des caractéristiques les plus avancées du moteur IDS. Si on les compare aux avions, SE est un avion à hélice, et IDS, un jet. Les deux sont des machines volantes, mais chacun a un usage approprié.

Le temps total d'installation est d'environ deux heures, si vous avez préparé votre environnement, ou plus, si vous ne l'avez pas fait. Il est extrêmement important de lire le *Guide de l'administrateur*, pour se familiariser avec les divers contrôles et options relatifs au moteur.

FAQ

Q. Puis-je utiliser les produits Microsoft avec une base de données Informix sur Linux ?

R. Bien sûr ! Tout produit Microsoft ou compatible Microsoft qui peut se connecter à une base de données ODBC peut se connecter à une base de données Informix sur Linux. Il suffit d'initialiser la connexion réseau entre le serveur Linux et le client Microsoft avant de se connecter.

Q. Puis-je connecter une base de données Informix sur Linux à une base de données Informix sur une autre plate-forme UNIX ?

R. Oui. Il faut configurer les paramètres réseau dans les fichiers de configuration appropriés et se relier à deux serveurs de bases de données différents pour travailler sur les deux simultanément. On peut aussi mettre en relation ces serveurs à l'aide de la réplication des données.

Q. Quel est le matériel nécessaire pour initialiser une base de données Informix ?

R. L'installation complète peut occuper approximativement 500 Mo pour un petit système, davantage pour une plus grande base de données. Informix peut fonctionner avec une mémoire aussi réduite que 8 Mo.

Q. Quel est le meilleur endroit pour trouver des informations sur Informix ?

R. Les FAQ Informix sont situées sur http://www.iiug.org, le site du groupe international des utilisateurs d'Informix. Outre la FAQ, on trouve sur ce site une pléthore d'outils gratuits utilisables avec la famille des produits Informix. Une bonne partie de l'histoire des moteurs Informix est décrite dans les FAQ. Ce site est une source riche d'outils, de sous-programmes, de programmes L4G, de programmes ESQLC, d'astuces et de techniques qui vous aident à faire fonctionner l'un des meilleurs moteurs de base de

© copyright Éditions Eyrolles

données. Vous pouvez télécharger ou déposer vos propres outils sur le site IIUG, et développer votre propre projet *open-source* avec d'autres développeurs.

Les FAQ vous aiguillent aussi vers d'autres ressources Informix, dont le réseau des développeurs Informix : http://www.informix.com/idn (IDN, *Informix Developer's Network*). C'est un site web destiné au développeur Informix sérieux, équipé de groupes de discussions, pour parler de sujets divers avec d'autres développeurs, des administrateurs ou d'autres participants. Il faut s'inscrire pour avoir accès à ce site web, mais il en vaut la peine, et il faut le visiter souvent pour se tenir au courant des mises à jour et des améliorations du logiciel.

© copyright Éditions Eyrolles

5

Installer et utiliser Sybase sur Linux

Dans ce chapitre

- Installation
- Configuration
- Test de la base de données
- Conception de la base de données
- Forces et faiblesses

© copyright Éditions Eyrolles

Introduction

Vers la mi-1998, j'ai mené, pour une organisation à but non lucratif, une mission de développement d'une base de données, administrée par l'Internet, d'objets d'art numérisés et catalogués. Cette organisation disposait d'un budget d'exploitation modeste (comme toute organisation de ce type) et n'était pas en mesure d'investir des sommes importantes dans une solution de base de données de grande marque : c'est pourquoi Linux apparut comme le choix évident. Après quelques semaines d'expérimentation des bases de données open-source et gratuites (*freeware*) pour Linux, sur quelques jeux de test, nous avons été fortement déçus par la lenteur des essais, quand nous avons atteint les dizaines de milliers d'enregistrements. La volumétrie totale devait passer de 200 000 enregistrements au départ à plusieurs millions : cette solution n'était pas envisageable et ne fut donc pas retenue. Découragés par les performances des bases de données open-source, nous avons été obligés de retenir NT, équipé de Microsoft SQL Server 6.5, comme plate-forme de base de données. Comme nous voulions conserver un serveur web fiable et rapide (Apache) et Perl sur Linux, nous avons démarré en utilisant le logiciel Open Client de Sybase et la bibliothèque Perl, écrite par Michael Peppler.

Heureusement, il semble que Microsoft ait disposé de la version initiale de SQL Server de Sybase, ce qui fait qu'Open Client pouvait encore se connecter. Nous n'étions pas satisfaits des performances de SQL Server sur NT mais, en même temps, nous n'étions pas en mesure de trouver une meilleure solution. Après environ un mois de développement, nous avons entendu des rumeurs selon lesquelles Sybase envisageait de porter la version 11.0.3 de SQL Server sur Linux. De surcroît, ils l'offraient gratuitement, et rien n'était perdu de notre mois de développement, puisque nous nous étions déjà servis de leur bibliothèque client. Quand Sybase a diffusé SQL Server, nous l'avons téléchargé et installé. Sans aucune modification, la base de données de développement que nous avions utilisée sur SQL Server NT fonctionnait près de 10 fois plus vite sous Sybase sur Linux. Nous avons été extrêmement satisfaits et nous nous sommes organisés pour développer en totalité le produit sur des logiciels gratuits. Depuis cette époque, Sybase a diffusé Adaptive Server Enterprise (ASE) 11.9.2 pour Linux, ainsi que beaucoup d'autres produits. ASE 11.9.2 est gratuit pour le développement, mais une licence est nécessaire pour le déploiement. Sybase offre toujours l'utilisation et le déploiement gratuits de SQL Server 11.0.3.

Dans ce chapitre, nous traiterons de l'installation de SQL Server Sybase sur une plate-forme Linux et indiquerons au moins quelques réglages. Comme il existe déjà beaucoup de bons ouvrages consacrés à l'exécution et au réglage de SQL Server Sybase, ce chapitre s'intéresse surtout à ce qui doit être fait pour que SQL Server fonctionne correctement sur Linux.

© copyright Éditions Eyrolles

Installation

Comme nous l'avons indiqué en introduction, Sybase propose actuellement deux versions de son serveur de base de données pour la plate-forme Linux. SQL Server Sybase 11.0.3.3 est disponible en libre téléchargement sur le site web de Sybase (http://www.sybase.com/products/databaseservers/linux/), gratuit à la fois pour le développement et le déploiement. Adaptive Server Enterprise (ASE) 11.9.2 est disponible en libre téléchargement sur le site de Sybase, mais la licence n'autorise à l'utiliser que sur une seule machine et seulement pour le développement. Pour déployer ASE 11.9.2 sur Linux, il faut une autre licence, payante. Ce chapitre se concentre sur la version 11.0.3.3, qui correspond bien à la nature de Linux parce qu'elle est gratuite, mais une grande part des informations s'appliquent aussi à la version 11.9.2.

Félicitations à Sybase pour avoir rendu ses serveurs de base de données sur Linux faciles à installer sur Red Hat Linux, en les distribuant au format RPM. Sybase fonctionne sur les autres versions de Linux dotées du noyau en version 2.0.36 ou supérieure, et de glibc2, mais l'utilisateur de ces systèmes devra télécharger et installer les utilitaires de gestion des paquets RPM à partir de http://www.rpm.org/.

Installer SQL Server 11.0.3

Pour installer SQL Server 11.0.3.3, il faut d'abord télécharger les RPM à partir du site web de Sybase ou les trouver dans le CD-ROM supplémentaire de Red Hat 5.2. Il faut se procurer les deux RPM suivants, qui contiennent :

- **Sybase-11.0.3.3-1.rpm :** les fichiers de programme ;
- **Sybase-docs-11.0.3.3-1.rpm :** la documentation.

Ces documents ne sont disponibles qu'en format Postscript, et sont assez longs à télécharger. Le RPM du CD-ROM Red Hat 5.2 porte le numéro de version 11.0.3-2.

Important. La version diffusée avec Red Hat Linux 5.2 a été à l'origine d'une certaine confusion. Dans cette version, Sybase 11.0.3 était ajouté au CD-ROM supplémentaire distribué avec la copie achetée de Red Hat 5.2. Le RPM sur le serveur web de Sybase était sybase-11.0.3-1.rpm, et celui du CD-ROM, sybase-11.0.3-2.rpm : ceux qui consultaient le site web ont donc été choqués de ne pas avoir la version la plus récente. En fait, la seule différence entre ces deux RPM était que la seconde — celle de Red Hat — ne lançait pas automatiquement le script d'installation après l'installation du RPM, car Red Hat avait demandé à Sybase de le reconfigurer parce que son outil RPM graphique n'acceptait pas ce script d'installation engendré automatiquement. La version disponible sur le site web de Sybase, à la rédaction de ce livre, est la 11.0.3.3. à utiliser de préférence car elle contient sans doute des corrections de bogues.

© copyright Éditions Eyrolles

Configurer la machine de destination

Avant de passer à l'installation des RPM, il est important d'examiner la question du partage des lecteurs physiques de la machine de destination. Beaucoup d'arguments ont été avancés, au sein du groupe de discussions sybase-linux sur Internet, à propos de la meilleure manière de partitionner et d'utiliser l'espace du disque dur de la machine. Nous décrirons ici les options qui ont fait l'objet du consensus général mais, si vous êtes intéressé par le débat, vous pouvez consulter les archives sur http://forum.swarthmore.edu/epigone/ase-linux-list.

Sybase sur Linux (comme beaucoup de programmes sur les machines UNIX) accède aux unités (*devices*) de base de données *via* le système de fichiers. C'est pourquoi SQL Server de Sybase ne gère pas directement les unités de base de données, qu'il s'agisse de fichiers d'un quelconque système de fichiers ou d'unités telles que des partitions d'un lecteur physique (par exemple **/dev/sda1**). Il y a des avantages et des inconvénients de part et d'autre.

Un fichier qui appartient au système de fichiers présente l'avantage de permettre un contrôle sur l'unité de base de données comme sur tout autre fichier de ce type : par exemple, le déplacer, créer des liens symboliques vers ce fichier ou le copier sur une autre machine. Avant que Sybase ne diffuse son logiciel de réplication pour Linux, c'était l'un des rares moyens faciles de répliquer une base de données. Malheureusement, sous le second système de fichiers étendu (le plus utilisé sur Linux), la taille des fichiers est limitée à deux giga-octets. Une unité de base de données, en tant que fichier du système de fichiers, est donc limitée à deux giga-octets.

Le passage par une partition de disque comme **/dev/sda1** sous Linux n'améliore pas la vitesse de la base de données de manière substantielle, comme on s'y attendrait, parce que Linux fait un usage intensif de la mémoire cache. Les accès à une partition de disque ou au système de fichiers se servent l'un comme l'autre du système de cache de Linux. Une partition de disque dédiée permet de disposer de fichiers de taille supérieure à deux giga-octets, ce qui est important pour les grandes bases de données. D'un autre côté, une partition dédiée à la base de données ne peut être atteinte que par l'unité de blocs correspondante et doit être allouée en partitionnant le disque.

L'un des principaux débats du groupe sybase-linux a porté sur l'usage de ces deux types de fichiers de base de données. Beaucoup de DBA sur d'autres plates-formes UNIX invoquent que la seule manière d'avoir une unité de base de données est de l'avoir sur une partition brute non formatée (*raw*). Ce type de partition est une unité du système pour laquelle le système d'exploitation n'utilise pas du tout le cache : toutes les écritures sont effectuées directement sur le disque, avant que le contrôle ne soit rendu au programme appelant. Les noyaux de Linux, jusqu'à la série 2.2, ne supportent pas les vraies unités *raw*,

© copyright Éditions Eyrolles

mais certains formats d'unités brutes seront probablement admis dans les versions ultérieures du noyau. Les administrateurs, qui pensent que ces unités sont nécessaires, avancent l'argument suivant à l'appui de leur conviction : comme Sybase maintient son propre cache et part de l'hypothèse que tout ce qui est écrit sur le disque est vraiment écrit quand il reprend le contrôle après une écriture, des incohérences peuvent apparaître en cas de coupure de l'alimentation. En clair, Sybase considère qu'il y a eu écriture sur le disque, alors que Linux n'a fait que conserver les données dans son cache, en vue de les écrire ultérieurement ; s'il y a une coupure avant que Linux n'ait réellement (matériellement) écrit, il est évident que cela produit des incohérences au niveau du système. Le meilleur argument que j'aie entendu contre cela postule que tout serveur de base de données dispose d'une alimentation ininterruptible (UPS, *uninterruptible power supply*). Si l'UPS avertit la machine d'une coupure, celle-ci a les moyens de s'assurer que la base de données et le reste du processus ont été arrêtés, en cas de rupture d'alimentation, avant que la batterie UPS ne soit épuisée.

Au sujet des possibilités en matière d'unités de données, voici quelques conseils. Si possible, dédier la machine entière à la base de données, et lui donner beaucoup de mémoire physique (autant que l'on peut). Sybase peut fonctionner avec 32 Mo de mémoire, s'il n'y a rien d'autre sur la machine, mais il y aura de nombreux accès aux disques, car il ne peut pas conserver assez d'informations dans le cache. Il est recommandé de ne pas utiliser moins de 128 Mo de mémoire dans un serveur de base de données. Evidemment, plus il y en a, meilleure est la performance. Si l'on envisage de faire simultanément des requêtes multiples sur la machine, il vaut mieux se tourner vers des processeurs multiples. Il est conseillé de ne pas faire fonctionner un serveur web et un serveur de base de données sur la même machine. On peut le faire, mais il faut savoir que la base de données sera beaucoup plus rapide et plus réactive si la machine n'est pas occupée à faire autre chose.

Pour toute base de données non triviale, il faudra consacrer au moins trois lecteurs SCSI au système. Ne pas utiliser de lecteurs IDE pour un serveur de base de données. Si l'on souhaite un système vraiment robuste, il vaut mieux prévoir au moins six lecteurs SCSI, dans une configuration RAID, ce qui permet de récupérer les données en cas de panne d'un périphérique. Le premier lecteur physique peut être assez petit (moins de 4 Go), car il ne contiendra que le système d'exploitation Linux, l'application de base de données et très peu d'autres choses. Le second lecteur (volume RAID) sera entièrement dédié aux journaux de transactions des bases de données. Le troisième peut alors être entièrement dédié aux unités de base de données. Si l'on souhaite exécuter des bases de données multiples sur une seule machine, il vaut mieux placer chaque base de données dans sa propre unité, sur un disque séparé : ainsi les requêtes n'obligeront pas le disque à faire des recherches sur deux unités situées sur le même disque.

© copyright Éditions Eyrolles

Partitionnez le premier disque SCSI en cinq partitions natives Linux au moins, et de préférence une partition d'échange Linux en plus. La première partition native accueillera le système de fichiers racine (*root*) et pourra être de taille assez petite (de 200 à 300 Mo) ; la seconde, le système de fichiers **/usr,** est d'environ 2 Go ; la troisième contiendra les fichiers **/opt**, où Sybase s'installera lui-même à partir du RPM. Cette partition sera au moins assez grande pour contenir la distribution Sybase installée (approximativement 300 Mo). Les deux dernières partitions natives Linux ne sont pas destinées à recevoir un système de fichiers et doivent avoir une taille d'environ 50 Mo ; elles contiendront les unités **master** et **sybsystemprocs** de Sybase. Les second et troisième lecteurs SCSI seront partitionnés comme les grandes partitions Linux natives (à la taille entière du disque), mais il ne faut pas y créer de système de fichiers.

La distribution Linux s'installe sur le premier lecteur SCSI.

Installer Sybase

Après la mise à jour de la machine et l'installation de Linux, telles que nous venons de les décrire, rassemblons quelques informations dont nous aurons besoin pour installer Sybase. D'abord, il faut savoir quelles unités correspondent aux partitions sur les lecteurs physiques déjà initialisés. Le programme **fdisk** affiche toutes les informations relatives aux partitions sur un lecteur physique. Voici un exemple d'utilisation du programme **fdisk** :

```
# fdisk /dev/sda
The number of cylinders for this disk is set to 1247.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): p
Disk /dev/sda: 255 heads, 63 sectors, 1247 cylinders
Units = cylinders of 16065 * 512 bytes
Device Boot Start End Blocks Id System
/dev/sda1 * 1 33 265041 83 Linux  mounted /
/dev/sda2 34 295 2100483 83 Linux  mounted /usr
/dev/sda3 296 336 321300 83 Linux  mounted /opt
/dev/sda4 337 1247 7309575 5 Extended
/dev/sda5 337 436 803250 83 Linux  master device
/dev/sda6 437 536 803250 83 Linux  sybsysprocs device
/dev/sda7 537 554 144585 82 Linux Swap  linux swap
/dev/sda8 555 1247 5558490 83 Linux  mounted /home


Command (m for help) : q
```

© copyright Éditions Eyrolles

Nous avons besoin de savoir quelles partitions ont été réservées pour la partition maître et la partition **sybsystemprocs.** Supposons que ce sont les partitions 5 et 6 du premier disque SCSI, correspondant respectivement aux fichiers spéciaux d'unités **/dev/sda5** et **/dev/sda6**. Nous devons aussi savoir quelles autres partitions ont été réservées sur le second et le troisième lecteur physique. Supposons que ce soient respectivement **/dev/sdb1** et **/dev/sdc1**.

Se connecter sur la console Linux en tant que root et installer les paquetages RPM par la commande suivante :

```
rpm –ivh sybase-11.0.3.3-1.rpm
```

Cette commande installera SQL Server Sybase dans le répertoire **/opt/sybase**. Après installation des fichiers, on retourne à l'invite de commande. Pour continuer l'installation, exécuter le script **/opt/sybase/install/sybinstall.sh**. Ce script affiche d'abord le texte de la licence à laquelle on doit souscrire, puis se poursuit par l'invite de l'installation et installe le serveur de base de données. La figure 5.1 présente les questions posées, suivies d'un exposé des options.

```
DO YOU AGREE WITH THE LISTED LICENSE CONDITIONS ? [Yes/No/List] : yes

Do you want this script to proceed by installing an account? [y/n]: y

Group sybase not found. Create it now? [y/n]: y

   Group and permissions for /opt/sybase are set !

User sybase not found. Create it now? [y/n]: y
Changing password for user sybase
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully

   User sybase is available, release user id is set !

   This completes your Sybase Adaptive Server Enterprise (ASE) package
   installation. To run Sybase ASE, you may like configuring your master
   and sybsystemprocs databases by continuing the installation procedure
   after logging in as sybase and executing /opt/sybase/install/sybinit.

   This action is the default when you login for the first time.
   Re-running this configuration procedure can be initiated by removing
   the. readme and. sybinit files from the 'sybase' installation
   directory.
```

Figure 5–1
Invite de sybinstall.sh

© copyright Éditions Eyrolles

La première question concerne la création d'un compte d'utilisateur pour le serveur de base de données Sybase. Il faut, en principe, laisser faire le script, de sorte que le serveur fonctionne en tant que compte d'utilisateur Sybase, ou de groupe. Il attend également la création d'un groupe et d'un utilisateur. Répondre oui à ces deux questions. Le script demande ensuite un mot de passe pour le compte utilisateur Sybase : choisir un mot de passe pour ce compte d'utilisateur et l'entrer à deux reprises. Le script met alors à jour la propriété des fichiers dans le répertoire **/opt/sybase,** de sorte que le groupe et le compte utilisateur que l'on vient de créer en soient propriétaires.

Le script signale que le fichier **/opt/sybase/install/sybinit** doit être exécuté avec les droits de l'utilisateur Sybase pour que la procédure d'installation suive son cours. Auparavant, il faut mettre à jour les propriétés de deux autres fichiers : en particulier, autoriser l'utilisateur et le groupe Sybase à écrire dans les partitions déjà réservées pour les unités de Sybase. Pour ce faire, exécuter, en tant que root, les commandes suivantes, dans lesquelles on remplacera **/dev/sdxx** par les unités de disque réelles à utiliser comme unités de base de données :

```
[root@wanderer install]# cd /dev
[root@wanderer /dev]# chown sybase:sybase /dev/sda5
[root@wanderer /dev]# chown sybase:sybase /dev/sda6
[root@wanderer /dev]# chown sybase:sybase /dev/sdb1
[root@wanderer /dev]# chown sybase:sybase /dev/sdc1
```

Maintenant que l'utilisateur Sybase a la propriété des fichiers qu'il utilisera comme unités de base de données, il faut s'assurer qu'aucun autre utilisateur n'a le droit de lire les partitions de la base de données. Si d'autres utilisateurs avaient accès aux unités, ils pourraient éventuellement lire les données des fichiers et reconstruire la base de données sans respecter la structure des autorisations Sybase. Les commandes suivantes assurent que seuls l'utilisateur et le groupe Sybase auront accès à ces fichiers :

```
[root@wanderer /dev]# chmod o-rwx /dev/sda5
[root@wanderer /dev]# chmod o-rwx /dev/sda6
[root@wanderer /dev]# chmod o-rwx /dev/sdb1
[root@wanderer /dev]# chmod o-rwx /dev/sdc1
```

Installer les logiciels clients optionnels

Il existe plusieurs paquetages logiciels clients qui facilitent considérablement l'administration et le développement de la plate-forme Sybase.

© copyright Éditions Eyrolles

SQSH

Le premier paquetage est une application X11, en lignes de commande, appelée *sqsh* (prononcer *squish*), disponible sur http://www.voicenet.com/gray/sqsh.html sous la licence générale publique GNU (GPL). C'est une application qui fournit une interface en lignes de commande, destinée à exécuter des instructions SQL, mais beaucoup plus agréable que le programme **isql** inclus dans Sybase – il vaut la peine de l'essayer quand on est administrateur d'un serveur, dans un shell SQL. On peut télécharger les sources et les paquetages binaires sur ce site.

Un conseil : récupérez le paquetage binaire et installez-le quelque part dans le chemin du système. Une fois installé, on peut l'utiliser pour se connecter à la base de données depuis la ligne de commande, au lieu de se servir de isql. Ce programme apporte de nombreuses améliorations à isql, par exemple :

- message de fin des commandes, avec édition de la ligne de commande ;
- variables ;
- redirection des commandes;
- travaux en tâches de fond ;
- contrôle des travaux ;
- historique.

Bibliothèques de développement

Les bibliothèques de développement constituent d'autres paquetages intéressants du logiciel client. Elles « font la soudure » entre le logiciel que l'on développe et le serveur Sybase. Sybase 11.0.3.3 va de conserve avec l'Open Client Library pour C mais, même si l'on ne développe pas en C, il se peut que l'on ait l'usage de ces bibliothèques.

Le développement avec Perl produit des scripts rapides et puissants qui permettent de modifier des éléments dans la base de données. Il y a actuellement deux interfaces Perl pour Sybase. La première, et la plus ancienne, est le paquetage *sybperl*, très proche de l'API C fournie avec l'Open Client Library pour les applications Perl. Le second paquetage est un pilote DBI qui correspond à l'interface DBI Perl. Ces paquetages présentent des avantages et des inconvénients. Le paquetage DBI permet de développer du code Perl qui peut établir des connexions à toutes les base de données dotées d'un pilote DBI (qui ressemble à ODBC ou JDBC). Cependant, comme DBI est générique, il y a certaines choses qu'il ne peut pas faire. Le paquetage **sybperl** donne toutes les possibilités dont on dispose sur un serveur Sybase avec l'Open Client Library de Sybase, mais il ne peut se connecter que sur un serveur Sybase.

© copyright Éditions Eyrolles

Ces deux paquetages sont disponibles sur les sites CPAN. Essayer http://www.cpan.org/, ou un site miroir.

Si l'on préfère développer en Java, Sybase propose un pilote JDBC écrit en pur Java, **jConnect,** sur son site http://www.sybase.com/products/internet/jconnect/. Ce pilote fonctionne sur toute plate-forme qui dispose d'une machine virtuelle Java.

Configuration

Nous pouvons maintenant poursuivre la configuration, comme la spécifie le script shell d'installation. Nous allons nous connecter en tant qu'utilisateur Sybase, puis exécuter l'application **/opt/sybase/install/sybinit.** Elle nous proposera plusieurs écrans de menus pour progresser dans l'installation.

Sur ma machine, équipée de Red Hat version 6.1, **sybinit** a des problèmes pour charger les fichiers de localisation. Je m'était servi de Red Hat 5.2 auparavant sans difficulté, il s'agit donc d'un nouveau problème introduit avec Red Hat 6.1., que j'ai essayé d'analyser. Ma conclusion est que les variables d'environnement **$LANG** et **$LC_ALL** sont à l'origine du problème. Il semble que Sybase ne comprenne pas la localisation **en_US** indiquée dans ces variables. Il suffit de désactiver ces variables pour que Sybase se serve de la valeur par défaut, **US_english**.

Une meilleure solution serait peut-être d'éditer le fichier texte **/opt/sybase/locales/locales.dat**. Ce fichier définit une méthode de représentation des chaînes locales dans LANG et LC_ALL, que Sybase comprend. Actuellement, il semble que Sybase a, sur la plate-forme Linux, ses propres définitions locales des chaînes **ENGLISH**, **FRENCH**, **GERMAN** et **JAPANESE**. Il suffit d'éditer ce fichier et de changer les lignes :

```
[linux]
        locale = ENGLISH, us_english, iso_1
        locale = FRENCH, french, iso_1
        locale = GERMAN, german, iso_1
        locale = JAPANESE, japanese, eucjis
        locale = default, us_english, iso_1
```

en :

```
[linux]
        locale = en_US, us_english, iso_1
        locale = ENGLISH, us_english, iso_1
        locale = fr_FR, french, iso_1
```

© copyright Éditions Eyrolles

```
locale = FRENCH, french, iso_1
locale = de_DE, german, iso_1
locale = GERMAN, german, iso_1
locale = ja_JP, japan, eucjis
locale = JAPANESE, japanese, eucjis
locale = C, us_english, iso_1
locale = default, us_english, iso_1
```

Ainsi les variables d'environnement $LC_ALL et $LANG sont autonomes. Évidemment, si ces variables contiennent une autre langue que celles qui sont spécifiées dans le fichier modifié, il faudra ajouter cette entrée au fichier, avec sa propre description de langue. Sybase 11.0.3.3 ne propose des traductions que pour le chinois, le français, l'allemand, le japonais, l'espagnol et l'américain mais, et c'est encore plus étrange, le fichier **locales.dat ne** contient de représentations que pour l'américain, le français, l'allemand et le japonais. On pourrait aisément en ajouter d'autres.

Configurer le serveur de base de données Sybase

Dans l'écran de la figure 5.2, choisir d'abord **3. Configure a Server product** en appuyant sur la touche 3.

```
SYBINIT

1.  Release directory:  /opt/sybase

2.  Edit / View Interfaces File

3.  Configure a Server product
4.  Configure an Open Client/Server product

Ctrl-a Accept and Continue, Ctrl-x Exit Screen, ? Help.

Enter the number of your choice and press return :
```

Figure 5–2
Écran initial de l'application sybinit

SQL Server et Backup Server seront configurés dans l'ordre pour que le serveur de base de données Sybase fonctionne correctement. Configurer d'abord SQL Server en appuyant sur la touche 1, puis le serveur de sauvegarde en appuyant sur la touche 2, dans l'écran de la figure 5.3.

Lorsqu'il s'agit d'une nouvelle installation de SQL Server, choisir **1. Configure a new SQL Server** dans l'écran de la figure 5.4. La même application (sybinit) pourra être utilisée ultérieurement, en choisissant **2. Configure**

© copyright Éditions Eyrolles

```
CONFIGURE SERVER PRODUCTS

Products:

    Product Date Installed Date Configured
1.  SQL Server 10 Sep 1998 15:47
2.  Backup Server 10 Sep 1998 15:47

Ctrl-a Accept and Continue, Ctrl-x Exit Screen, ? Help.

Enter the number of your choice and press return :
```

Figure 5–3
Écran de choix du produit (sybinit)

```
NEW OR EXISTING SQL SERVER

1.  Configure a new SQL Server
2.  Configure an existing SQL Server
3.  Upgrade an existing SQL Server

Ctrl-a Accept and Continue, Ctrl-x Exit Screen, ? Help.

Enter the number of your choice and press return :
```

Figure 5–4
Écran de configuration de SQL Server (sybinit)

an existing SQL Server, pour modifier les options que nous allons maintenant initialiser.

À cette étape de l'installation, il faut donner un nom au serveur SQL Server (figure 5.5). Ce nom est simplement un nom logique qui indique aux clients de cette machine à quel SQL Server physique se connecter réellement (adresse IP et numéro de port). Si l'on envisage d'exécuter des instances multiples de SQL Server sur la même machine (ou si l'on partage le fichier de configuration des interfaces entre des serveurs et des clients multiples), les serveurs doivent avoir des noms différents. Beaucoup de clients utilisent **SYBASE,** en tant que nom de serveur SQL Server par défaut, pour se connecter : on peut ainsi laisser le nom de SYBASE. Pour changer le nom, choisir **1** et entrer le nouveau nom, puis appuyer sur **Ctrl-A** pour continuer.

Les actions proposées à la figure 5.6 s'effectuent dans l'ordre de la liste. La première étape est d'ajouter une entrée au fichier des interfaces. Sybase se sert de ce fichier texte (situé dans **/opt/sybase/interfaces**) pour indiquer aux clients et au serveur l'adresse IP et le port à utiliser. Il contient la correspondance entre les noms logiques et les paramètres, de telle sorte que les clients puissent les utiliser pour se connecter aux serveurs.

© copyright Éditions Eyrolles

```
ADD NEW SQL SERVER

1.  SQL Server name:  SYBASE

Ctrl-a Accept and Continue, Ctrl-x Exit Screen, ? Help.

Enter the number of your choice and press return :
```

Figure 5–5
Écran de création d'un nouveau SQL Server (sybinit)

```
SQL SERVER CONFIGURATION

1.  CONFIGURE SERVER'S INTERFACES FILE ENTRY Incomplete

2.  MASTER DEVICE CONFIGURATION Incomplete
3.  SYBSYSTEMPROCS DATABASE CONFIGURATION Incomplete
4.  SET ERRORLOG LOCATION Incomplete
5.  CONFIGURE DEFAULT BACKUP SERVER Incomplete

6.  CONFIGURE LANGUAGES Incomplete
7.  CONFIGURE CHARACTER SETS Incomplete
8.  CONFIGURE SORT ORDER Incomplete

9.  ACTIVATE AUDITING Incomplete

Ctrl-a Accept and Continue, Ctrl-x Exit Screen, ? Help.

Enter the number of your choice and press return :
```

Figure 5–6
Écran de configuration de SQL Server (sybinit)

```
SERVER INTERFACES FILE ENTRY SCREEN

    Server name:  SYBASE

1.  Retry Count:  0
2.  Retry Delay:  0

3.  Add a new listener service

Ctrl-a Accept and Continue, Ctrl-x Exit Screen, ? Help.

Enter the number of your choice and press return :
```

Figure 5–7
Écran de configuration du fichier d'interfaces de SQL Server (sybinit)

© copyright Éditions Eyrolles

Les deux premières options de l'écran (figure 5.7) spécifient respectivement combien de fois un client peut essayer de se reconnecter s'il n'a pas réussi la première fois, et l'attente entre deux tentatives de connexion. On peut sans problème laisser cette valeur à 0. En sélectionnant **3. Add a new listener service** dans l'écran de la figure 5.7, on peut spécifier à quel port et quel nom d'hôte le serveur sera attaché, ce qui spécifie un autre alias de nom logique pour le serveur.

```
EDIT TCP SERVICE

1.  Hostname/Address: wanderer.deftech.com
2.  Port:
3.  Name Alias:

4.  Delete this service from the interfaces entry

Ctrl-a Accept and Continue, Ctrl-x Exit Screen, ? Help.

Enter the number of your choice and press return :
```

Figure 5–8
Écran de configuration du fichier d'interfaces du TCP Service Listener de SQL Server (sybinit)

Dans l'écran de la figure 5.8, on indique un numéro de port que le serveur Sybase va écouter. Ce numéro de port sera supérieur à 1024, car le serveur fonctionne comme un utilisateur normal et non comme root. Il est recommandé d'utiliser le port 1433, parce que c'est le port par défaut dont se sert Microsoft SQL Server. Cela facilite l'accès des clients Microsoft SQL Server sous Windows au serveur SQL Server Sybase (il y a quelques restrictions à l'utilisation de ces clients, car les développements de Sybase et de Microsoft SQL Server ont divergé depuis que Microsoft a acquis la licence initiale du code auprès de Sybase).

Le champ **Hostname/address** sera renseigné soit par le nom d'hôte, soit par l'adresse IP de l'interface sur laquelle SQL Server écoutera. SQL Server Sybase écoutera seulement sur cette interface. Le champ **Name alias** permet d'attribuer un autre nom logique à SQL Server. Ce nom logique spécifie cette entrée d'interface particulière.

Quand ces options sont configurées, appuyer sur **Ctrl-A** pour revenir à l'écran de saisie du fichier d'interfaces de SQL Server. On peut ajouter une autre entrée Listener dans le fichier d'interfaces si l'on veut que le même SQL Server écoute sur une autre interface ou port.

Quand on a renseigné toutes les interfaces Listener que l'on veut, appuyer sur **Ctrl-A** et répondre oui à l'invite de sauvegarde des modifications du fichier d'interfaces.

© copyright Éditions Eyrolles

```
## WANDERER on wanderer.deftech.com
##       Services:
##              query tcp     (1433)

WANDERER
        query tcp ether wanderer.deftech.com 1433

## LOCALSERVER on localhost
##       Services:
##              query tcp     (1433)

LOCALSERVER
        query tcp ether localhost 1433

## SYBASE on wanderer.deftech.com
##       Services:
##              query tcp     (1433)
##              query tcp     (1433)
##              master tcp     (1433)
##              master tcp     (1433)

SYBASE
        query tcp ether wanderer.deftech.com 1433
        query tcp ether localhost 1433
        master tcp ether wanderer.deftech.com 1433
        master tcp ether localhost 1433
```

Figure 5–9
Fichier d'interfaces créé par l'application sybintit

La figure 5.9 présente un fichier d'interfaces créé par l'application sybinit, avec deux interfaces Listener. On peut voir que l'une est sur l'interface **wanderer.deftech.com** avec le port 1433 et l'autre sur l'interface localhost, avec le même port 1433. Les deux interfaces ont des noms logiques différents — **WANDERER** et **LOCALSERVER**. Comme ce fichier est un fichier texte normal, il est modifiable directement, en dehors de sybinit.

Après en avoir terminé avec le fichier d'interface du serveur, sélectionner, dans l'écran de la figure 5.6, le menu **2. Master device configuration** pour choisir l'emplacement de l'unité **master**. Spécifier le fichier qui contiendra l'unité **master** (**/dev/sda5** dans cet exemple) et la taille de l'unité. Elle aura une taille d'au moins 21 Mo, mais ce doit être la taille exacte de la partition créée auparavant, si l'on se sert d'une unité. Appuyer sur **Ctrl-A** pour continuer. L'application sybinit émet un avertissement qui indique que le fichier ou l'unité choisi n'est pas recommandé pour une unité de serveur : il se réfère aux différences, expliquées à la section précédente, entre unités par blocs et unités brutes sur la machine de destination. Appuyer sur **Entrée** pour continuer.

© copyright Éditions Eyrolles

Ensuite, on initialise l'unité de base de données **sybsystemprocs** comme on l'a fait pour l'unité **master**. La seule différence est que l'on choisit **/dev/sda6** comme emplacement de l'unité.

L'option 4 (figure 5.6) spécifie le fichier dans lequel SQL Server Sybase écrira ses messages d'erreur. Ce peut être un fichier quelconque du système de fichiers, pourvu que l'utilisateur Sybase y ait accès en écriture. Il sera vérifié périodiquement, car toutes les erreurs rencontrées par SQL Server y seront consignées.

L'option 5 (figure 5.6) spécifie le nom logique du serveur de sauvegarde. Ce peut être un serveur de sauvegarde Sybase existant, indiqué dans le fichier d'interface, ou le nom d'un nouveau serveur de sauvegarde à créer lors de la configuration du serveur de sauvegarde. Si c'est un serveur autonome, indiquer seulement son nom et s'en souvenir pour être capable de l'indiquer plus tard, lors de la configuration du serveur de sauvegarde. Son nom par défaut est **SYB_BACKUP**.

L'option 6 (figure 5.6) fixe les encodages de langue et la langue par défaut que l'on souhaite installer dans cette base de données.

L'option 7 (figure 5.6) choisit les jeux de caractères et le jeu de caractères par défaut à installer dans la base de données.

L'option 8 (figure 5.6) choisit la méthode de classement qui s'applique à la base de données. Attention : ce choix s'applique aux instructions SORT BY et à l'opérateur « = ». Donc, si vous choisissez, par exemple, un classement de caractères indifférents à la casse, *ABC*, *Abc*, *abc* seront égaux dans toutes les instructions SQL de comparaison.

L'option 9 (figure 5.6) détermine si l'audit est autorisé dans la base de données. Ce sujet déborde du cadre de ce chapitre, mais le paquetage sybase-docs.11.0.3.3.rpm contient des informations suffisantes à propos de cette option. Dans l'exemple, elle n'est par activée (choix par défaut).

Quand les neuf options ont été passées en revue, appuyer sur **Ctrl-A** pour continuer. Sybinit demande si l'on veut exécuter la configuration de SQL Server tout de suite. Répondre oui : sybinit initialise les unités de base de données **master** et **sybsystemprocs** et démarre le serveur de base de données. Il émet le même avertissement que précédemment (sur les unités par blocs) pour l'unité **master**. Là encore, ignorer cet avertissement et appuyer sur **Entrée**. Si tout se passe bien, le serveur de base de données Sybase sera installé et fonctionnera à la fin de ce processus. La figure 5.10 montre le résultat. Remarquer, à la deuxième ligne, que l'unité **master** n'est pas **/dev/sda5**, mais un fichier normal **/opt/sybase/master.dat**. C'est parce qu'il ne s'agit, dans cet exemple, que d'une machine de développement et de démonstration, et non d'une vraie base de données.

© copyright Éditions Eyrolles

```
Execute the SQL Server Configuration now ? y

WARNING: '/opt/sybase/master.dat' is a regular file which is not
recommended for a Server device.
Press <return> to continue.
Running task: create the master device.
Building the master device.
Done
Task succeeded: create the master device.
Running task: update the SQL Server runserver file.
Task succeeded: update the SQL Server runserver file.
Running task: boot the SQL Server.
waiting for server 'SYBASE' to boot...
Task succeeded: boot the SQL Server.
Running task: create the sybsystemprocs database.
sybsystemprocs database created.
Task succeeded: create the sybsystemprocs database.
Running task: install system stored procedures..
.....................................................................
.....................................................................
.....................................Done

Task succeeded: install system stored procedures.
Running task: set permissions for the 'model' database.
Done
Task succeeded: set permissions for the 'model' database.
Running task: set the default character set and/or default sort order
for the SQL Server.
Setting the default character set to iso_1
Sort order 'binary' has already been installed.
Character set 'iso_1' is already the default.
Sort order 'binary' is already the default.
Task succeeded: set the default character set and/or default sort order
for the SQL Server.
Running task: set the default language.
Setting the default language to us_english
Language 'us_english' is already the default.
Task succeeded: set the default language.

Configuration completed successfully.
Press <return> to continue.
```

Figure 5–10
Processus d'installation et de mise en route du serveur de base de données Sybase

© copyright Éditions Eyrolles

Configurer le serveur de sauvegarde Sybase

Quand le serveur de base de données fonctionne, configurer le serveur de sauvegarde (figure 5.3). Si l'on a installé une baie de serveurs[1], on n'aura besoin que d'un serveur de sauvegarde pour tous les serveurs de base de données. Pour ce faire, ajouter le nom logique du serveur de sauvegarde dans le fichier d'interfaces de chacun des serveurs de base de données et initialiser le serveur de sauvegarde par défaut comme on l'a vu à l'écran de la figure 5.6, à l'étape 5. S'il s'agit d'un serveur autonome ou d'un serveur de sauvegarde spécifique de la ferme, il faudra créer un nouveau serveur de sauvegarde sur la machine.

Choisir de créer un nouveau serveur de sauvegarde et lui donner un nom logique. On utilisera le même nom que celui du serveur de sauvegarde par défaut, dans le serveur de base de données (figure 5.3). Quand le nom est choisi, appuyer sur **Ctrl-A** et le programme sybinit affichera l'écran de la figure 5.11.

```
BACKUP SERVER CONFIGURATION

1.  Backup Server errorlog:  /opt/sybase/install/backup.log
2.  Enter / Modify Backup Server interfaces file information
3.  Backup Server language:  us_english
4.  Backup Server character set:  iso_1
5.  Backup Server tape configuration file: /opt/sybase/backup_tape.cfg

Ctrl-a Accept and Continue, Ctrl-x Exit Screen, ? Help.

Enter the number of your choice and press return :
```

Figure 5–11
Écran de configuration du serveur de sauvegarde d'applications sybinit

L'option 1 permet de choisir le fichier dans lequel les messages du serveur de sauvegarde seront conservés : un fichier quelconque du système de fichiers convient. L'option 2 crée, dans le fichier d'interfaces, une entrée pour le serveur de sauvegarde : elle ressemble à l'entrée relative au serveur de base de données. On peut spécifier le nombre d'essais, le délai entre essais et le service Listener du serveur de sauvegarde. On doit spécifier au moins un service Listener : ce peut être un port ou une interface quelconques. Les options 3 et 4 spécifient la langue et le jeu de caractères que l'on veut utiliser sur le serveur de sauvegarde. L'option 5 indique le fichier de configuration qui contient des informations sur les bandes. Ce sujet n'est pas abordé ici,

1. Le terme anglais *farm* (« ferme »), quelque peu inhabituel en ce domaine, est adéquat : il désigne bien un ensemble d'entités, différentes les unes des autres mais d'économie semblable, qui prennent tout leur sens au travers d'une relation dynamique à l'ensemble. *N.d.T.*

© copyright Éditions Eyrolles

mais est traité dans le *Guide de l'administrateur système* de la documentation des RPM, ainsi que sur le site web de Sybase à http://sybooks.sybase.com.

Après avoir renseigné les cinq options, appuyer sur **Ctrl-A** et sybinit propose de démarrer le serveur de sauvegarde. Répondre oui. Si tout se passe bien, le serveur de sauvegarde sera mis en service. La figure 5.12 montre la trace du démarrage du serveur de sauvegarde par sybinit.

```
Execute the Backup Server configuration now ? y
Running task: update the Backup Server runserver file.
Task succeeded: update the Backup Server runserver file.
Running task: boot the Backup Server.
waiting for server 'SYB_BACKUP' to boot...
Task succeeded: boot the Backup Server.

Configuration completed successfully.
Press <return> to continue.
```

Figure 5–12
Processus de démarrage initial du serveur de sauvegarde

Configurer les bibliothèques client-serveur de Sybase

Après les serveurs, il faut aussi configurer les bibliothèques client. Cette étape consiste essentiellement en sélections de menus, car il n'y a aucun paramètre à fournir. L'option **4. Configure an Open Client/Server product**, présentée dans l'écran de la figure 5.2, propose trois produits client à configurer ; configurer chacun d'eux. Quand c'est terminé, appuyer sur **Ctrl-A** plusieurs fois pour sortir de l'application sybinit.

Beaucoup d'applications client s'attendent à ce que la variable d'environnement **$SYBASE** soit placée dans la racine de l'arbre d'installation Sybase, et leur indique où trouver le fichier d'interfaces ainsi que d'autres fichiers de données. Si de nombreux utilisateurs de votre machine se connectent de la même manière en tant que clients, depuis cette machine ou d'autres machines Linux, insérez la ligne suivante dans **/etc/profile**, de manière à initialiser cette variable pour tout le monde :

```
export SYBASE=/opt/sybase
```

Attention : certains scripts et serveurs, qui accèdent au serveur *via* les bibliothèques client C ou Perl, vont aussi rechercher cette variable d'environnement. Si les scripts CGI sybperl ou les applications clientes ne peuvent pas se connecter au serveur, vérifiez qu'ils disposent bien de la variable **$SYBASE** dans leur environnement.

© copyright Éditions Eyrolles

Démarrer automatiquement le serveur de base de données et le serveur de sauvegarde

À ce stade, le serveur de base de données et le serveur de sauvegarde fonctionnent. Pour qu'un serveur de base de données soit indépendant, on souhaite, en général, que ces serveurs démarrent en même temps que la machine. L'installation de Sybase a créé un fichier **/etc/rc.d/init.d/sybase**, un script de démarrage **rc** compatible avec la méthode standard de Red Hat de démarrage et d'arrêt des services. Pour que Sybase démarre au moment du boot et que tout se passe bien quand on l'arrête, il faut créer des liens symboliques vers ce fichier dans **/etc/rc.d/rc0.d**, **/etc/rc.d/rc3.d**, **/etc/rc.d/rc5.d** et **/etc/rc.d/rc6.d**, dont la signification sort du cadre de ce chapitre. Le jeu de commandes qui suivent assure que Sybase démarre et se termine correctement. Ces commandes doivent être exécutées sous le compte root.

```
#cd /etc/rc.d/rc0.d
#ln –s. ./init.d/sybase K05sybase
#cd /etc/rc.d/rc3.d
#ln –s. ./init.d/sybase S95sybase
#cd /etc/rc.d/rc5.d
#ln –s. ./init.d/sybase S95sybase
#cd /etc/rc.d/rc6.d
#ln –s. ./init.d/sybase K05sybase
```

Après exécution de ces commandes, toutes les bases de données Sybase et les serveurs de sauvegarde de la machine seront démarrés au prochain boot.

Modifier le mot de passe de l'administrateur système

Le serveur de base de données Sybase est opérationnel. On peut se connecter au serveur depuis une machine locale, à l'aide des noms logiques accessibles dans le fichier des interfaces. Sybase propose une application appelée *isql*, qui permet d'envoyer de manière interactive des instructions SQL au serveur de base de données et de recevoir les résultats à l'écran. NB : on peut aussi télécharger *sqsh*, un outil beaucoup plus agréable pour des fonctionnalités similaires. Voir sa description à la section « SQSH », page 209.

Sybase installe le serveur de base de données avec un mot de passe vide pour l'administrateur système (**sa**), ce qui représente un risque important ; par conséquent, il faut le changer aussi vite que possible. Pour changer le mot de passe de **sa**, il faut se connecter au serveur avec l'outil isql et exécuter la procédure stockée **sp_password,** comme suit :

© copyright Éditions Eyrolles

```
[brad@wanderer bard]$ /opt/sybase/bin/isql -U sa -S WANDERER
Password:
1> sp_password @new_password="INOTELL"
2> go
Password correctly set.
(return status = 0)
```

Normalement, quand on met à jour le mot de passe d'un utilisateur, on doit aussi donner les paramètres **caller_password**, **new_password** et **loginame** à la procédure stockée **sp_password.**

Configurer les unités de bases de données et les bases de données

La prochaine étape de configuration du serveur est de créer les unités de bases de données où seront stockées les tables utilisateur. Dans l'exemple, il ne faudra qu'une seule base de données utilisateur sur la machine, mais on peut très facilement avoir des bases de données multiples sur chaque unité et des unités multiples sur la machine, ainsi qu'une base de données unique répartie sur des unités multiples. Sybase offre toutes ces possibilités, qui sont autant d'extensions logiques du cas traité ici.

Pour créer une nouvelle unité de base de données, utiliser la commande **disk init**, dont la syntaxe est :

```
DISK INIT NAME="<name>",
    PHYSNAME="<device>",
    VDEVNO=<devnumber>,
    SIZE=<size>
    [, CNTRLTYPE=<controller>] [, VSTART=<offset>] [, CONTIGUOUS]
```

Pour créer une unité en tant que fichier du système de fichiers, il suffit de spécifier dans **PHYSNAME** le chemin du fichier à créer ; pour créer une unité sur une unité Linux (telle qu'une partition d'un lecteur physique), indiquer dans **PHYSNAME** le chemin du fichier virtuel de l'unité ; le paramètre **NAME** de **disk init** est un nom de chaîne unique qui référence l'unité de base de données sur ce serveur et **VDEVNO** le numéro d'unité de la nouvelle unité : il devra être supérieur au numéro le plus élevé connu du serveur. On peut trouver les unités en cours d'utilisation et leurs numéros avec la procédure stockée **sp_helpdevice. SIZE** est la taille de la nouvelle unité, un nombre entier de blocs de 2048 octets. Quand on indique la taille d'une unité de base de données qui va occuper une partition, il faut s'assurer qu'elle est bien égale à la taille de la partition elle-même, car on ne peut avoir qu'une seule unité de base de données sur une partition.

© copyright Éditions Eyrolles

Par exemple, on exécutera les deux commandes suivantes dans isql :

```
1> disk init name="xact", physname="/dev/sdb1",
2> VDEVNO=2, SIZE=<SIZE OF PARTITION in 2KB blocks>
3> go
1> disk init name="data1", physname="/dev/sdc1",
2> VDEVNO=3, SIZE=<SIZE OF PARTITION in 2KB blocks>
3> go
```

Maintenant, deux nouvelles unités sont disponibles pour les bases de données utilisateur de SQL Server. La première, appelée **xact**, contiendra les journaux de transaction. Il est préférable de placer les journaux de transaction sur un fichier physique séparé de la base de données : si l'un des fichiers physiques est abîmé, on peut reconstruire la base de données à partir de l'autre fichier. La seconde unité, appelée **data1**, contiendra les tables de données de la base de données. Elle se nomme **data1** car c'est la première créée : on y ajoute des unités de base de données (ou des lecteurs physiques) au fur et à mesure que la base de données grandit.

La prochaine étape consiste à créer réellement des bases de données utilisateur, qui résideront dans les nouvelles unités : pour ce faire, utiliser la commande SQL de création de base de données. Cette commande doit être exécutée avec une connexion à la base de données **master** (taper **use master**). La syntaxe de cette commande est :

```
create database database_name
[on {default | database_device } [= size ]
   [, database_device  [= size ]...]
[log on database_device  [ = size  ]
   [, database_device  [= size ]]...]
[with override]
[for load]
```

La clause **on** spécifie sur quelle unité sera placée la base de données, et la clause ***=size*** sa taille, en méga-octets. La clause **log on** indique à SQL Server où placer le journal des transactions. Par exemple, pour créer une base de données, **user_data**, de manière à ce qu'elle occupe les deux unités **data1** et **xact**, passer la commande :

```
1> create database user_data on data1=<SIZE OF PARTITION in MB>
2> log on xact=<SIZE OF PARTITION in MB>
3> go
CREATE DATABASE: allocating 51200 pages on disk 'data1'
CREATE DATABASE : allocating 14336 pages on disk 'xact'
```

© copyright Éditions Eyrolles

Nous disposons maintenant d'une très grande base de données utilisateur appelée **user_data**. Pour la mettre en service, passer la commande **use user_data** et créer des tables, des vues, des procédures stockées, etc.

Créer des comptes utilisateur et des autorisations

Il n'est évidemment pas recommandé de se servir du nom d'utilisateur **sa** pour toute connexion d'un client à la base de données. On peut créer d'autres utilisateurs, d'autres noms de connexion et d'autres groupes, et spécifier les autorisations des différents utilisateurs et groupes.

Les noms de connexion (*login*) ne sont pas la même chose que les utilisateurs (*users*). Les noms de connexion ne servent qu'au cours du processus de connexion réelle à la base de données. Ensuite, le nom d'utilisateur réel peut changer, selon la base de données à laquelle l'utilisateur se connecte. Certains noms de connexion peuvent avoir le même nom d'utilisateur et d'autres disposer d'un alias dans différentes bases de données. Certains noms de connexion peuvent même ne donner accès à aucune base de données. Les nouveaux noms de connexion sont créés dans la base de données **master** par la procédure stockée **sp_addlogin,** qui attend deux paramètres : le nom de connexion et le mot de passe. Par exemple, créons deux noms de connexion dans la base de données **user_data** :

```
1> sp_addlogin brad, <password>
2> go
Password correctly set.
Account unlocked.
New login created.
(return status = 0)
1> sp_addlogin mary, <password>
2> go
Password correctly set.
Account unlocked.
New login created.
(return status = 0)
```

Une fois ce nom créé, on peut spécifier la base de données par défaut à laquelle la connexion se fait. Si aucune base de données par défaut n'est spécifiée, l'utilisateur de ce nom sera placé dans la base de données **master.** En général, ce n'est pas ce que l'on souhaite, parce qu'un utilisateur normal ne doit pas avoir le droit de modifier cette base de données. On utilise la procédure stockée **sp_defaultdb** dans la base de données **master** pour spécifier dans quelle base de données un utilisateur sera placé immédiatement après la connexion.

© copyright Éditions Eyrolles

Utilisateurs et groupes

Dans les grandes bases de données, où l'on peut répartir les utilisateurs en un nombre quantifiable de groupes ayant un ensemble de droits précis sur les objets de la base de données, il est intéressant de créer des groupes et d'attribuer les autorisations aux groupes. Cela permet d'ajouter ou d'enlever facilement un nom d'utilisateur au système sans avoir à se souvenir des autorisations particulières ni à mettre à jour chacun des objets auxquels l'utilisateur a accès.

C'est particulièrement vrai en matière de gestion du personnel : les employés sont interchangeables et, à ce titre, ne peuvent offrir de garantie à l'entreprise. Il en résulte qu'ajouter ou supprimer des utilisateurs dans une base de données représente vite un travail d'informaticien à plein temps. Il y aura aussi, invariablement, le cas du nouvel employé qui aura vraiment besoin de certains accès pour mener à bien un projet critique, mais n'aura pas les autorisations voulues. Si cette personne est classée dans un type particulier d'utilisateur (par exemple les « ventes »), il suffira de créer un nouvel utilisateur faisant partie du groupe « ventes » pour que toutes les autorisations applicables à ce groupe lui soient attribuées.

Par exemple, les noms de connexion créés ci-dessus seront placés dans la base **user_data**, avec les commandes suivantes, exécutées dans isql :

```
1> use master
2> go
1> sp_defaultdb brad, user_data
2> go
Default database changed.
(return status = 0)
1> sp_defaultdb mary, user_data
2> go
Default database changed.
(return status = 0)
```

Les groupes sont des ensembles d'utilisateurs qui ont les mêmes autorisations. En utilisant les groupes, on peut spécifier des autorisations qui reposent sur le type des utilisateurs et, quand le nombre d'utilisateurs augmente, ajouter de nouveaux utilisateurs aux groupes, sans avoir à changer les autorisations sur tous les objets de la base de données. Pour créer un nouveau groupe dans une base de données, se placer dans la base de données et exécuter la procédure stockée **sp_addgroup.** Cette procédure, qui n'a qu'un paramètre, le nom du nouveau groupe, crée le groupe. Pour créer un groupe nommé **editors** dans la base de données **user_data**, on utilisera la commande :

© copyright Éditions Eyrolles

```
1> use user_data
2> go
1> sp_addgroup editors
2> go
```

Dans une base de données, un utilisateur peut, sans que ce soit obligatoire, avoir le même nom qu'un nom de connexion. Par exemple, le propriétaire de la base de données est un utilisateur spécial nommé **dbo**, mais le nom de connexion **sa** est le **dbo** de toutes les bases de données. Pour ajouter un nom de connexion qui sera **dbo** d'une base de données particulière, ajouter d'abord ce nom comme il a été décrit auparavant, puis créer un alias de l'utilisateur **dbo**. On le fait avec la procédure stockée **sp_addalias :** le nom de connexion spécifié correspond à l'utilisateur spécifié pour la base de données courante. Cette même procédure stockée donne tous les alias d'un utilisateur. Ainsi, pour spécifier que le nom de connexion **brad** sera **dbo** de la base de données **user_data**, on exécutera les commandes suivantes dans isql, en tant qu'administrateur (**sa**) :

```
1> use user_data
2> go
1> sp_addalias brad, dbo
2> go
Alias user added.
(return status = 0)
```

D'autres utilisateurs de la base de données peuvent être ajoutés à l'aide de la procédure stockée **sp_adduser** : elle indique le nom de connexion, le nouveau nom d'utilisateur dans cette base de données, et le groupe auquel appartient l'utilisateur. Par exemple, **mary** est un nouvel utilisateur à ajouter à la base de données **user_data.** Mary est éditeur, aussi nous en ferons un membre du groupe des éditeurs et elle héritera toutes les autorisations des éditeurs :

```
1> use user_data
2> go
1> sp_adduser mary, mary, editors
2> go
New user added.
(return status = 0)
```

À la limite, seul le dbo a l'autorisation de faire n'importe quoi dans une base de données. À chaque objet (table, vue, procédure stockée ou colonne) de la base de données est associé un ensemble d'autorisations, pour chaque utilisateur ou groupe. Le dbo a le droit d'accorder des autorisations aux autres utilisateurs de la base de données pour certaines actions sur certains objets. La procédure d'attribution d'autorisations à un utilisateur ou à un groupe leur donne — ou enlève — cette autorisation (sélection, mise à jour, insertion,

© copyright Éditions Eyrolles

suppression, références ou exécuter) sur un objet particulier. Voici la syntaxe des instructions **GRANT** et **REVOKE** correspondantes :

```
grant {all [privileges]| permission_list}
on { table_name [(column_list)]
    | view_name[(column_list)]
    | stored_procedure_name}
to {public | name_list | role_name}
[with grant option]

revoke [grant option for]
{all [privileges] | permission_list}
on { table_name [(column_list)]
    | view_name [(column_list)]
    | stored_procedure_name}
from {public | name_list | role_name}
[cascade]
```

Par exemple, ces commandes accordent au groupe **editors** le droit de lire les données des tables **auteurs**, **chapitres** et **livres** — il faut, bien sûr, créer d'abord ces tables :

```
1> grant select on auteurs to editors
2> grant select on chapitres to editors
3> grant select on livres to editors
4> go
```

Tester la base de données

Récapitulons : nous avons une base de données complètement opérationnelle qui contient une base de données utilisateur appelée **user_data**, et deux noms de connexion : **brad** et **mary**. Actuellement, ces deux noms de connexion[2] démarrent par défaut dans la base de données **user_data,** mais **mary** (et le groupe **editors** tout entier) n'a aucune autorisation (sauf si l'on a créé les tables auteurs, chapitres et livres, et accordé l'autorisation de les lire aux éditeurs).

Nous allons maintenant parcourir quelques étapes de test élémentaire de l'installation de la base de données.

Exécutons d'abord la commande suivante, pour vérifier que le serveur de base de données fonctionne :

2. Rappel : *login* ; on dit aussi « compte ».

© copyright Éditions Eyrolles

```
# /etc/rc.d/init.d/sybase status

USER   PID %CPU %MEM VSZ   RSS   TTY   STAT START TIME COMMAND
sybase 648 0.0  8.5  21764 10904 pts/0 S    18:26 0:02
     /opt/sybase/bin/dataserver -d/opt/sybase/master.dat -sSYBASE
     -e/opt/sybase/install/errorlog -i/opt/sybase

sybase 649 0.0  3.1  6720  4088  pts/0 S    18:26 0:00
     /opt/sybase/bin/backupserver -SSYB_BACKUP
     -e/opt/sybase/install/ backup.log -I/opt/sybase/interfaces
     -M/opt/sybase/bin/sybmultbuf -Lus_english -Jiso_1
     -c/opt/sybase/backup_tape.cfg
```

On voit apparaître une ligne par serveur de base de données et par processeur dont on dispose. La sortie fait état de la machine qui exécute un serveur de base de données appelé **SYBASE** et d'un serveur de sauvegarde appelé **SYB_BACKUP**. Si seule la première ligne apparaît (USER PID...), cela signifie que le serveur de base de données Sybase ne fonctionne pas. Pour le démarrer, exécuter la commande :

```
/etc/rc.d/init.d/sybase start
```

Exécuter ensuite la commande **sybase status,** une seconde fois, pour vérifier s'il fonctionne. Si ce n'est pas le cas, vérifier le journal d'erreurs dans le fichier indiqué à sybinit. Le fichier par défaut est **/opt/sybase/install/errorlog.** Remédier à toutes les erreurs signalées et redémarrer le serveur avec la commande **sybase start**.

Quand on a la certitude que le serveur de base de données fonctionne, on peut s'y connecter comme l'un des trois utilisateurs qui ont été créés. Pour se connecter au serveur, depuis la ligne de commande, exécuter la commande :

```
# /opt/sybase/bin/isql -U <username> -S <servername>
```

et entrer le mot de passe. On doit alors voir l'invite **1>**. Elle indique que l'on peut envoyer une nouvelle requête à la base de données et que l'on est sur la première ligne de la requête. On peut entrer une requête en SQL avec autant de retours à la ligne que l'on veut : chaque fois que l'on appuie sur la touche **Entrée**, le numéro s'incrémente de un dans l'invite. Quand on est satisfait de la commande, on tape **go** sur une nouvelle ligne puis on appuie sur la touche **Entrée**. Isql envoie alors la requête au serveur et affiche les résultats à l'écran.

Important. Beaucoup de commandes Sybase produisent une sortie plus large que 80 colonnes. Si on les exécute dans une fenêtre xterm ou une console qui dispose de 120 colonnes au moins, la sortie sera beaucoup plus lisible. Dans certains exemples qui suivent, le code a été modifié légèrement, pour obtenir une présentation en colonnes qui en facilite la lecture.

© copyright Éditions Eyrolles

Essayons quelques commandes simples pour interrompre le shell SQL. D'abord, faisons la liste de tous les objets de la base de données que nous pouvons utiliser. Sybase dispose pour cela d'une procédure stockée, **sp_help** qui, sans aucun paramètre, listera tous les objets (tables, vues, procédures stockées, etc.) de la base de données courante. Si l'on donne en paramètre le nom d'un objet particulier, elle affichera des informations relatives à cet objet. Par exemple, si l'on est dans la base de données **user_data (**se connecter avec le login **brad**, ou avec **sa**, et exécuter la commande **use user_data)** et que l'on exécute **sp_help**, on obtient une liste telle que :

```
1> sp_help
2> go
 Name            Owner           Object_type
 --------------- --------------- ----------------------
 sysalternates   dbo             system table
 sysattributes   dbo             system table
 syscolumns      dbo             system table
 syscomments     dbo             system table
 sysconstraints  dbo             system table
 sysdepends      dbo             system table
 sysgams         dbo             system table
 sysindexes      dbo             system table
 syskeys         dbo             system table
 syslogs         dbo             system table
 sysobjects      dbo             system table
 syspartitions   dbo             system table
 sysprocedures   dbo             system table
 sysprotects     dbo             system table
 sysreferences   dbo             system table
 sysroles        dbo             system table
 syssegments     dbo             system table
 systhresholds   dbo             system table
 systypes        dbo             system table
 sysusermessages dbo             system table
 sysusers        dbo             system table

 User_type       Storage_type    Length Nulls Default_name Rule_name
 --------------- --------------- ------ ----- ------------ ---------

(return status = 0)
```

Elle montre tous les objets (ici, les tables système) de la base de données, ainsi que leur propriétaire. On peut alors exécuter **sp_help** avec l'une de ces tables en paramètre et l'on obtient la liste des colonnes de la table. Par exemple, si l'on exécute **sp_help sysusers**, on obtient la liste que voici :

© copyright Éditions Eyrolles

```
1> sp_help sysusers
2> go
Name                         Owner                          Type
---------------------------- ------------------------------ -------------
sysusers                     dbo                            system table

Data_located_on_segment        When_created
------------------------------ -------------------------
system                         Jan 1 1900 12:00AM

Column_name    Type         Length Prec Scale Nulls Default_name  Rule_name Identity
-------------- ------------ ------ ---- ----- ----- ------------ --------- --------
suid           smallint        2    NULL NULL      0 NULL         NULL       0
uid            smallint        2    NULL NULL      0 NULL         NULL       0
gid            smallint        2    NULL NULL      0 NULL         NULL       0
name           sysname        30    NULL NULL      0 NULL         NULL       0
environ        varchar       255    NULL NULL      1 NULL         NULL       0

index_name  index_description                     index_keys index_max_rows_per_page
----------- ------------------------------------ ---------- --------------------
sysusers    clustered,unique located on system       suid                  0
ncsysusers1 nonclustered,unique located on system    name                  0
ncsysusers2 nonclustered,unique located on system    uid                   0

(3 rows affected)
keytype  object         related_object  object_keys           related_keys
-------- -------------- --------------- --------------------- -------------------
foreign  sysalternates  sysusers        suid,*,*,*,*,*,*,*    suid,*,*,*,*,*,*,*
common   sysobjects     sysusers        uid,*,*,*,*,*,*,*     uid,*,*,*,*,*,*,*
common   sysprotects    sysusers        uid,*,*,*,*,*,*,*     uid,*,*,*,*,*,*,*
common   systypes       sysusers        uid,*,*,*,*,*,*,*     uid,*,*,*,*,*,*,*
primary  sysusers       -- none --      suid,*,*,*,*,*,*,*    *,*,*,*,*,*,*,*
Object is not partitioned.

(return status = 0)
```

Elle nous donne toutes sortes d'informations dont nous n'avons pas nécessairement besoin, mais nous indique les colonnes de la table et leur type. Elle indique également quels sont les index de cette table, et sur quelles colonnes portent ces index.

Commençons à créer nos propres objets en créant une nouvelle table. Soit, par exemple, un site web utilisateur qui autorisera les connexions avec un nom d'utilisateur et un mot de passe. Il faut évidemment connaître les utilisateurs à un instant donné, ainsi que leur mot de passe. Nous sommes aussi, probablement, intéressés par certaines informations personnelles telles que leur nom, leur adresse postale et leur adresse e-mail.

Pour ranger toutes ces informations et les utiliser facilement, nous créons la table suivante :

© copyright Éditions Eyrolles

```
1> CREATE TABLE user_logins (
2>     username VARCHAR(20) PRIMARY KEY,
3>     password VARCHAR(20),
4>     fname VARCHAR(30),
5>     lname VARCHAR(50),
6>     email VARCHAR(100)
7> )
8> go
```

Après avoir exécuté ce code, à l'aide de isql, dans la base de données **user_data,** nous disposons maintenant d'un nouvel objet appelé **user_logins**. Si vous êtes curieux, essayez encore **sp_help** pour afficher tous les objets et voir ce que l'on obtient à propos de **user_logins**. Ajoutons maintenant quelques données à cette table :

```
INSERT INTO user_logins (username, password, fname, lname, email) VALUES
('brad', 'let me in', 'Brad', 'Dietrich', 'brad@deftech.com')
INSERT INTO user_logins VALUES ('mary', 'let me in', 'Mary', 'Poppins',
'mary@somewhere.net')
INSERT INTO user_logins VALUES ('brad', 'password', 'Brad', 'Poppins',
'brad@somewhere.net')
```

Remarquez que l'on obtient une erreur en ajoutant un autre utilisateur **brad** avec le même nom d'utilisateur. C'est parce que nous avons ajouté à **username** une clause **PRIMARY KEY**, qui implique que ce champ soit unique. Sybase refuse d'insérer la même clé deux fois dans ce champ.

Nous allons maintenant extraire quelques informations de la base de données, par exemple, la liste de tous les utilisateurs, triée sur le nom. La commande suivante nous sert à cela :

```
SELECT username, lname, fname, email FROM user_logins SORT BY lname
```

L'opération de tri sur cette table prendrait un temps assez long si la table était bien remplie, car nous n'avons pas déclaré d'index sur cette colonne. Un index sur une colonne rend les tris et les références à la colonne incroyablement plus rapides. Une bonne conception de base de données exige de décider quelles colonnes indexer. On ne peut pas se permettre d'indexer chaque colonne car cela augmente la charge sur la base de données, et joue à la fois sur la taille (pour conserver les arbres d'index) et sur la vitesse (pour mettre à jour les arbres d'index quand les tables changent). Pour créer un index sur les colonnes **lname** et **fname** de sorte que le rapport précédent soit plus rapide, on exécute la commande :

```
CREATE INDEX user_logins_lname_fname ON user_logins(lname, fname)
```

© copyright Éditions Eyrolles

Finalement, pour supprimer cette table qui ne nous intéresse pas vraiment — ce n'est qu'un test — on passe la commande :

```
DROP TABLE user_logins
```

Si toutes ces commandes ont fonctionné à la fois pour l'utilisateur **sa** et pour l'utilisateur **brad**, votre base de données est complètement configurée et correcte. Si des erreurs se sont produites pendant ce test simple, revenez en arrière dans le chapitre, assurez-vous que vous avez bien terminé toutes les tâches de configuration et d'installation décrites. Maintenir une base de données qui fonctionne de manière impeccable est souvent un travail à temps plein pour un DBA accompli ; mais de petites bases de données pourront être maintenues et fonctionner grâce aux commandes exposées dans ce chapitre.

Conception d'une base de données

La conception est une étape très importante dans tout projet de développement d'une base de données. C'est souvent la première étape du développement, et peut-être l'une des plus critiques. Si une base de données est mal pensée dès le départ, elle risque malheureusement de rester en l'état, car tout changement implique de refaire tous les développements qui reposent sur elle.

Quand on s'apprête à développer une nouvelle base de données, il ne suffit pas d'imaginer les types de données qui doivent y être rangés, mais aussi toutes les dépendances et les cas d'utilisation qui peuvent se présenter maintenant ou à l'avenir. Concevoir une base de données revient à essayer de prédire le futur, car toute base de données non triviale est appelée à durer. Il faut donc prendre en considération la teneur des requêtes auxquelles elle devra répondre, ainsi que la manière dont elle répondra, en termes de structures.

Pour essayer de donner une esquisse de ce qui entre en ligne de compte dans le développement d'une bonne base de données, considérons, par exemple, la société qui publie cet ouvrage. Elle peut créer une base de données d'informations sur tous les livres qu'elle publie, les chapitres de chacun des livres, et les auteurs de ces chapitres. Dans une première tentative de définition de cette base de données, prenons en compte les données qui sont référencées par d'autres. Évidemment, un livre contiendra des chapitres ; nous représenterons cela en incluant la clé primaire du livre dans la colonne **book** du chapitre, ce qui nous permettra de savoir aisément à quel livre appartiennent les chapitres.

Le prochain ensemble de données à relier est l'auteur du chapitre. Il est possible (c'est le cas de ce livre) que les différents chapitres du livre soient écrits par des auteurs différents. Donc, notre modèle de données doit pouvoir accep-

© copyright Éditions Eyrolles

ter un auteur par chapitre. Le seul moyen de le faire sera d'inclure la clé primaire de la table **auteurs** dans la table **chapitres**. Cela fonctionne tant qu'il n'y a pas plus d'un auteur par chapitre. Mais, comme il est à prévoir que plusieurs auteurs participent à l'écriture d'un même chapitre, ce modèle de données sera inadapté. De même, nous ne devons pas inclure la clé primaire du chapitre dans la table **auteurs**, parce qu'il est probable qu'un auteur écrira plusieurs chapitres.

Cette relation n à n apparaît souvent dans la modélisation des données. Pour modéliser cette relation n à n correctement, nous pouvons retenir l'approche naïve qui consiste à créer des colonnes auteur multiples dans la table des chapitres, mais nous nous trouverons invariablement face au cas où le nombre de colonnes sera insuffisant. De même, écrire une instruction SQL qui gère le cas des colonnes clés étrangères multiples sera, au mieux, fastidieux.

La meilleure manière de modéliser cette relation n à n est de créer une table de liaison qui contiendra les clés des tables **chapitres** et **auteurs**. Donc, pour chaque chapitre qu'un auteur écrira, il y aura une ligne dans la table de liaison, et pour chaque chapitre qui a des auteurs multiples, il y aura une ligne par auteur dans la table de liaison.

Pour éclairer cet exemple, le script SQL suivant va créer les quatre tables, les remplir de quelques données d'essai et faire des requêtes. Supposons qu'il n'y ait qu'un seul titre « Le modèle de base de données Wild ». Il aura deux chapitres intitulés « Chapitre 1 » et « Chapitre 2 ». Chapitre 1 a été écrit exclusivement par **Joe User** et le chapitre 2, par **Joe User** et **Sally User**.

```
CREATE TABLE books (
  book_id NUMERIC(10,0) PRIMARY KEY,
  book_title VARCHAR(100)
)

CREATE TABLE chapters (
  chapter_num NUMERIC(10,0) PRIMARY KEY,
  book_id NUMERIC(10,0) REFERENCES books(book_id),
  chapter_title VARCHAR(100)
)

CREATE TABLE authors (
  author_id NUMERIC(10,0) PRIMARY KEY,
  fname VARCHAR(30),
  lname VARCHAR(30)
)

CREATE TABLE chapter_author_link (
  link_id NUMERIC(10,0) IDENTITY PRIMARY KEY,
  chapter_num NUMERIC(10,0) REFERENCES chapters(chapter_num),
  author_id NUMERIC(10,0) REFERENCES authors(author_id)
)
```

© copyright Éditions Eyrolles

```
INSERT INTO books VALUES (1, 'The Wild Database Model')
INSERT INTO chapters VALUES (1, 1, 'Chapter 1')
INSERT INTO chapters VALUES (2, 1, 'Chapter 2')
INSERT INTO authors VALUES (1, 'Joe', 'User')
INSERT INTO authors VALUES (2, 'Sally', 'User')
INSERT INTO chapter_author_link (chapter_num, author_id)
VALUES (1, 1)
INSERT INTO chapter_author_link (chapter_num, author_id)
VALUES (2, 1)
INSERT INTO chapter_author_link (chapter_num, author_id)
VALUES (2, 2)

SELECT book_title, chapter_num, chapter_title, lname, fname
FROM books, chapters, authors, chapter_author_link
WHERE books.book_id=chapters.book_id
AND chapters.chapter_num=chapter_author_link.chapter_num
AND authors.author_id=chatper_author_link.author_id

book_title              chapter_num chapter_title lname fname
----------------------- ----------- ------------- ----- -----
The Wild Database Model 1           Chapter 1     User  Joe
The Wild Database Model 2           Chapter 2     User  Joe
The Wild Database Model 2           Chapter 2     User  Sally
```

Problèmes

Cette section traite de quelques problèmes qui surviennent quand on se sert de Sybase. Certains d'entre eux sont génériques de Sybase et affectent toutes les plates-formes, d'autres sont spécifiques de la version Linux de Sybase.

Colonne Identité (auto-incrémentée)

Sybase offre la possibilité de créer des colonnes de type entier qui s'incrémentent automatiquement chaque fois que l'on insère une ligne dans la table. Ces colonnes sont intéressantes pour générer des clés primaires, dans les tables qui n'ont pas naturellement de colonnes contenant des données uniques, ou si l'on veut référencer une colonne particulière d'une autre table, sans se servir d'une longue chaîne en clé primaire.

Une colonne identité doit être du type de données NUMERIC et sera toujours un entier. La taille n'est pas importante, mais on prend des risques à se servir d'un NUMERIC trop grand. Pour créer une colonne identité, on utilise la syntaxe suivante :

© copyright Éditions Eyrolles

```
CREATE TABLE <tablename> (
   table_id NUMERIC(<size>, 0) IDENTITY PRIMARY KEY,
   ...
)
```

Si l'on décide d'utiliser des colonnes identité, il faut être prudent. D'abord, Sybase ne permet pas d'insérer ses propres valeurs dans la colonne identité d'une table, sauf si on le spécifie à l'avance. Pour indiquer cette option à Sybase, exécuter la commande suivante :

```
set identity_insert <table_name> on
```

Après avoir exécuté cette commande, on peut insérer des valeurs dans la colonne identité, tout comme dans n'importe quelle colonne NUMERIC. Remettre l'option **identity_insert** à off pour cette table quand on en a terminé. Quand **identity_insert** est remise à off, Sybase redémarre la production de valeurs pour les colonnes identité à une valeur générée supérieure à la valeur maximum de la colonne courante.

Si l'on utilise des colonnes identité dans Sybase pour des valeurs de temps, on peut remarquer qu'à partir d'un certain point la colonne identité commence à contenir des nombres vraiment très grands, avec un écart important entre les nombres successifs. Cela est dû au fait que Sybase se sert d'une optimisation appelée le *moment critique de l'identité*. Quand on insère la première ligne dans une table avec une colonne identité, Sybase réserve en réalité un ensemble de nombres pour les affecter plus rapidement lors d'insertions ultérieures — c'est le moment critique de l'identification. En effet, si la base de données est arrêtée avec l'option **nowait**, ou bien en cas d'urgence, Sybase ne sait pas si tous les numéros de l'ensemble ont déjà été utilisés. Par conséquent, il redémarre en allouant un autre ensemble de nombres, d'où l'écart important dans la série des numéros assignés. On peut résoudre ce problème en éditant le fichier **SYBASE.cfg** dans **/opt/sybase** et en affectant une faible valeur au facteur de moment critique, mais la seule manière d'éviter que ce problème ne se produise est de s'assurer que la base de données est toujours arrêtée correctement (voir, plus loin, les scripts **rc** de Sybase).

Un dernier point important : la variable globale, bien spéciale, **@@IDENTITY**. C'est une variable globale, affectée à chaque session, qui contient toujours la dernière valeur d'identité assignée lors de la dernière insertion : elle donne donc le nombre qui a été assigné en dernier, à l'occasion d'une insertion dans une table ayant une colonne d'identité.

Voici un exemple illustré par la figure 5.13.

© copyright Éditions Eyrolles

```
1> CREATE TABLE table_a (
2>    table_a_id NUMERIC(10,0) IDENTITY PRIMARY KEY,
3>    value_a VARCHAR(30)
4> )
5> go
1> CREATE TABLE table_b (
2>    table_b_id NUMERIC(10,0) IDENTITY PRIMARY KEY,
3>    table_a_id NUMERIC(10,0) REFERENCES table_a(table_a_id),
4>    value_b VARCHAR(30)
5> )
6> go
1> INSERT INTO table_a (value_a) VALUES ('value_a')
2> INSERT INTO table_b (table_a_id, value_b) VALUES (@@IDENTITY, 'value_b')
3> go
(1 row affected)
(1 row affected)
1> SELECT * FROM table_a, table_b WHERE table_a.table_a_id=table_b.table_a_id
2> go
table_a_id      value_a          table_b_id      table_a_id      value_b
------------    ------------------  -------------  ------------------------------
1               value_a          1               1                value_b

(1 row affected)
```

Figure 5–13
Exemple de session isql qui illustre l'usage de la variable @@IDENTITY

Conformité à SQL

Un problème très important : Sybase 11.0.3.3 ne supporte pas directement la syntaxe JOIN dans la clause FROM des instructions SQL. La plupart des nouvelles bases de données permettent de joindre deux tables liées par une relation de clé primaire ou étrangère dans une clause JOIN. Sybase propose la même fonctionnalité qu'une clause de jointure, mais celle-ci s'appuie sur les opérateurs *=, = et =* dans la clause WHERE. Cette limite n'est pas un vrai défaut, mais elle engendre une certaine gêne lors de la migration d'une base de données d'une plate-forme à une autre. Toutes les instructions SQL à migrer doivent être converties dans une syntaxe à base d'opérateurs au lieu de la syntaxe de JOIN.

La figure 5.14 présente une base de données imaginaire qui garde l'historique des auteurs de ce livre, chapitre par chapitre. Comme on le voit, il n'y a que deux tables, l'une appelée **auteurs** et l'autre **chapitres**. La table des chapitres définit un lien référentiel vers la table des auteurs, de manière à ce que chaque chapitre corresponde à un auteur défini dans la table des auteurs.

© copyright Éditions Eyrolles

```
CREATE TABLE authors (
    lname VARCHAR(30) PRIMARY KEY,
    fname VARCHAR(30),
    initial CHAR(1)
)
CREATE TABLE chapters (
    chapter_number NUMERIC(10,0) PRIMARY KEY,
    author_lname VARCHAR(30) REFERENCES authors(lname),
    title VARCHAR(100)
)
```

Figure 5–14
Code SQL pour créer deux tables liées

```
INSERT INTO authors (fname, lname, initial) VALUES ('Brad', 'Dietrich', 'W')
INSERT INTO chapters (chapter_number, author_lname, title)
VALUES (6, 'Dietrich', 'Sybase')
```

Figure 5–15
Code SQL pour remplir les tables

Dans la figure 5.15, on renseigne la table auteurs avec le nom de l'auteur d'un chapitre, et la table des chapitres avec le titre du chapitre et l'auteur du chapitre. Si nous n'utilisons pas, pour **author_lname**, une valeur pour laquelle nous aurons déjà créé une ligne dans la table des auteurs, Sybase émet un message d'erreur. C'est l'effet souhaité quand on stipule que la colonne se réfère à une autre colonne de la table des auteurs. La plupart des bases de données actuelles utilisent la syntaxe **JOIN** dans la clause **FROM** de l'instruction **SELECT** pour retrouver les informations de ces deux tables. L'instruction de la figure 5.16 est l'instruction qui servirait dans la plupart des bases de données :

```
SELECT chapter_number, title, lname, fname
FROM chapters INNER JOIN authors ON chapters.author_lname=authors.lname
WHERE chapter_number=6
```

Figure 5–16
Code SQL pour retrouver le titre et l'auteur d'un chapitre en utilisant la syntaxe JOIN

Malheureusement, ce n'est pas le cas de Sybase en version 11.0.3.3 ; la figure 5.17 montre comment formuler cette instruction dans Sybase pour parvenir au même résultat.

```
SELECT chapter_number, title, lname, fname
FROM chapters, authors
WHERE chapters.author_lname=authors.lname AND chapter_number=6
```

Figure 5–17
Code SQL pour retrouver le titre et l'auteur d'un chapitre en utilisant la syntaxe de Sybase

© copyright Éditions Eyrolles

On voit que la phrase de la figure 5.17 a exactement la même signification, mais qu'elle est formulée autrement. La syntaxe JOIN a été adoptée pour regrouper toutes les relations entre les tables dans la clause FROM, de sorte qu'il est plus facile de lire l'instruction sous cette forme SQL et de comprendre ce qui se passe. Le tableau 5.1 montre la correspondance entre les opérateurs de Sybase et la syntaxe JOIN utilisée dans les autres SGBDR.

Tableau 5–1
Correspondance entre la syntaxe de JOIN et les opérateurs de Sybase

Syntaxe de jointure	Opérateur de Sybase
INNER JOIN	=
LEFT JOIN	*=
RIGHT JOIN	=*

Environnement d'exécution

On trouve une description vraiment fouillée des réglages de Sybase SQL Server dans la documentation des paquetages RPM, ainsi que sur le site web de Sybase http://sybooks.sybase.com/. Il y a notamment deux sujets que l'on se doit d'aborder dans ce chapitre : cela améliorera sensiblement les performances de Sybase SQL Server pour Linux, et vous épargnera des migraines.

Utilisation de la mémoire par Sybase

Il faut indiquer à Sybase combien de mémoire le système pourra utiliser. Par défaut, il ne se sert que d'environ 16 Mo de mémoire pour ses caches internes et les données. On obtient de bien meilleures performances si l'on donne une valeur un peu plus grande. La règle de base est de soustraire la quantité de mémoire physique dont on a besoin pour les autres processus qui s'exécutent sur la machine (assez peu, espérons-le) de la mémoire physique totale. Le résultat indique la taille de la mémoire à donner à Sybase. Attention, en sens inverse, à ne pas allouer trop de mémoire à Sybase, sinon Linux sera forcé d'éliminer quelque chose sur le disque quand il aura besoin de mémoire physique, ce qui tue absolument les performances d'un serveur de base de données.

On peut spécifier la quantité de mémoire que Sybase essaiera d'utiliser, dans la ligne « total memory » du fichier **SYBASE.cfg** dans le répertoire **/opt/sybase**. Le nombre à indiquer est un nombre de blocs de 2 Ko que Sybase utilisera pour la mémoire partagée.

© copyright Éditions Eyrolles

Il reste que la mise à jour de cette valeur pose un léger problème. Sybase a été développé pour fonctionner sur des machines SMP, et donc il alloue toujours une grande part de la mémoire au titre de la mémoire partagée, de sorte que tous ses propres processus (sur chaque processeur) puissent accéder à sa mémoire. Linux, dans sa configuration par défaut, n'autorise pas les segments de mémoire partagée à dépasser la taille d'environ 32 Mo. Heureusement, avec l'avènement du noyau 2.2.x de Linux, cette valeur par défaut peut être changée lors du boot au travers du fichier système **/proc/sys.** Pour stipuler un segment de mémoire partagée plus grand, exécuter la commande suivante :

```
echo 134217728 > /proc/sys/kernel/shmmax
```

Elle fait passer la taille maximale du segment de mémoire partagée à 128 Mo. Le maximum théorique que Linux autorise est 1 Go.

Important. Attribuer à la mémoire partagée une valeur aussi élevée annihile un espace d'adressage précieux sur les serveurs de plus de 2 Go de RAM ; utiliser plutôt un vrai matériel 64 bits.

Il faut ajouter ce qui suit, quelque part dans vos scripts **rc**, de sorte que cela se produise lors du boot, avant que le serveur Sybase ne soit démarré : ajouter ceci dans le script **/etc/rc.d/init.d/sybase** de sorte qu'il se lit comme :

```
start)
    echo –n "Setting Shared Memory Max to 128MB ... "
    echo 134217728 > /proc/sys/kernel/shmmax
    echo "done"
    echo -n "Starting Sybase ASE ... "
    ( cd ${SYBASE}/install ;
      unset LANG ; unset LC_ALL ;

      for run_server in RUN_* ; do
          su -c "startserver -f ${run_server}" sybase > /dev/null 2>&1 ;
          echo -n "${run_server} " ;
      done )
    echo
     ;;
```

Scripts rc de Sybase

Malheureusement, le script **rc** fourni par Sybase (**/etc/rc.d/init.d/sybase**) provoque l'arrêt de Sybase SQL Server, avec **nowait,** quand on lui demande de s'arrêter. Si l'on se sert de colonnes identité (colonne entière à incrémentation automatique de Sybase), cela peut être cause d'ennuis dus à de grands écarts entre les valeurs de ces colonnes. La solution à ce problème est de

© copyright Éditions Eyrolles

changer le script **rc** pour arrêter Sybase SQL Server avec l'option **wait**, au lieu de **nowait**. Voici ce qu'il faut modifier :

```
 stop)
     echo -n "Shuting down Sybase ASE configured servers: "
     killproc dataserver
     killproc backupserver
# Add your own openserver applications.
     echo
      ;;
```

que l'on change en :

```
 stop)
     echo -n "Stopping Sybase SQL Server"
     /opt/sybase/bin/isql -Usa -P******* -S$DSQUERY <<EOSQL
shutdown SYB_BACKUP with nowait
go
shutdown
go
exit
EOSQL
      echo
       ;;
```

C'est une menace potentielle pour la sécurité, puisque le mot de passe **sa** est stocké dans ce fichier. Bien protéger ce fichier des indiscrets par la commande **chmod 700 /etc/rc.d/init.d/sybase**. Seul l'utilisateur root pourra exécuter ce fichier, et aucun autre utilisateur ne pourra le lire, l'écrire ou l'exécuter.

Tempdb

Sybase se sert d'une base de données appelée **tempdb,** qui gère toutes les tables temporaires dont il se sert pour produire les requêtes complexes telles que les GROUP BY, ORDER BY ou les jointures. Cette base de données, dans la configuration par défaut de Sybase SQL Server, est une base de données de 2 Mo sur l'unité master. Je trouve personnellement cette base de données trop petite (j'ai rencontré des erreurs liées au manque d'espace) et trop lente. Beaucoup de gens recommandent de la placer sur un disque RAM pour la rendre plus rapide, de sorte qu'elle puisse dépasser la taille de 2 Mo.

Le noyau par défaut de Red Hat 6.1 a une taille maximale par défaut, celle des disques RAM de 4 Mo. Si vous souhaitez avoir un tempdb sur un disque RAM de plus de 4 Mo, il faut éditer le fichier **/etc/lilo.conf** et ajouter la ligne suivante à la section du noyau de démarrage, avant d'exécuter la commande **lilo** pour réinstaller LILO :

© copyright Éditions Eyrolles

```
append="ramdisk_size=<size in KB>"
```

Supposons que l'on ait spécifié, dans **/etc/lilo.conf**, la taille du disque RAM à utiliser pour la base de données tempdb, on peut envisager d'ajouter cette unité à la base de données Sybase. Par conséquent, pour que **/dev/ram** (le disque RAM) soit une unité de la base de données, exécuter la commande suivante dans isql (sa).

```
1> disk init name="tempdbdev", physname="/dev/ram", vdevno=XX,
size=<size of partition in 2k blocks>
    (Note : remplacer XX par la valeur correcte de vdevno !)
2> go
```

On a maintenant créé une nouvelle unité de base de données qui se sert de **/dev/ram**. Il ne faut utiliser cette unité que pour la base de données **tempdb.** Elle n'existe que dans la RAM et ne doit rien contenir d'utile lors du redémarrage de la machine. Puisque c'est le cas, par définition, de tempdb (Sybase crée tempdb à partir du modèle de la base de données chaque fois qu'il démarre), on peut facilement utiliser la RAM comme support mémoire de tempdb.

Maintenant, il faut encore transférer tempdb dans cette unité **tempdbdev.** Le jeu de commandes qui suit crée un fichier disque de 20 Mo et transfère tempdb depuis l'unité master vers cette unité.

```
# /opt/sybase/bin/isql -U sa
Password:
1> disk init name="tempdbdev", physname="/dev/ram", vdevno=4, size=10000
    (Note: remplacer =10000 par le nombre de blocs de 2ko que vous
     avez spécifié dans ramdisk_size dans le fichier /etc/lilo.conf file)
2> go

1> alter database tempdb on tempdbdev=20
    (Note: remplacer =20 par le nombre de méga-octets que vous
     avez spécifié pour tempdbdev dans disk init)
2> go
Extending database by 9984 pages on disk tempdbdev

1> use tempdb
2> go
1> sp_helpdb tempdb
    (Note: cette commande n'affiche que l'utilisation de la base de données)
2> go
name        db_size     owner     dbid      created             status
----------- ----------- --------- --------- ------------------- ---------------------
tempdb      21.5 MB     sa        2         Jan 12, 2000        select into/bulkcopy
```

© copyright Éditions Eyrolles

```
device_fragments               size            usage           free kbytes
------------------------------ --------------- --------------- -----------
master                         2.0 MB          data and log    1184
tempdbdev                      19.5 MB         data and log    19968

 device                        segment
 ------------------------------ ------------------------------------------------
 master                        default
 master                        logsegment
 master                        system
 tempdbdev                     default
 tempdbdev                     logsegment
 tempdbdev                     system
(return status = 0)

1> sp_dropsegment "default", tempdb, master
2> go
DBCC execution completed. If DBCC printed error messages, contact a user with
System Administrator (SA) role.
Segment reference to device dropped.
(return status = 0)

1> sp_dropsegment "logsegment", tempdb, master
2> go
DBCC execution completed. If DBCC printed error messages, contact a user with
System Administrator (SA) role.
DBCC execution completed. If DBCC printed error messages, contact a user with
System Administrator (SA) role.
Could not update the last-chance threshold for database tempdb
Segment reference to device dropped.
(return status = 0)

1> sp_dropsegment "system", tempdb, master
2> go
DBCC execution completed. If DBCC printed error messages, contact a user with
System Administrator (SA) role.
Segment reference to device dropped.
WARNING: There are no longer any segments referencing device 'master'.  This
device will no longer be used for space allocation.
(return status = 0)

1> sp_helpdb
2> testdb
3> go
The specified database does not exist.
(return status = 1)

1> sp_helpdb
2> tempdb
3> go
```

© copyright Éditions Eyrolles

```
 name        db_size     owner       dbid        created             status
-------------  -------------  -----------  ------      --------------      --------
tempdb      21.5 MB     sa          2           Jan 12, 2000        select into/bulkcopy

 device_fragments                size            usage               free kbytes
------------------------------  -------------   -------------------  -----------
master                          2.0 MB          data only           1232
tempdbdev                       19.5 MB         data and log        19888

 device                         segment
 ------------------------------ ----------------------------------------------
 master                         -- unused by any segments --
 tempdbdev                      default
 tempdbdev                      logsegment
 tempdbdev                      system

(return status = 0)
```

Remarquer le message d'avertissement du dernier **sp_dropsegment** : c'est exactement ce que nous voulons (aucune des bases de données de tempdb n'utilise plus l'unité master) aussi ne faut-il pas s'effrayer. Une dernière remarque à propos du transfert de tempdb sur un disque RAM : il faut se souvenir que la RAM que l'on alloue à tempdb doit être soustraite de la RAM totale, avant de déterminer la quantité de RAM dont Sybase disposera pour son cache interne (voir la section précédente sur l'utilisation de la mémoire).

Résumé

Ce chapitre a présenté l'installation et la configuration de Sybase SQL Server sur Red Hat Linux 6.1. Sybase est un fidèle partisan de la communauté Linux et l'a prouvé, en lui offrant ce serveur quelque peu ancien et en lui proposant son serveur actuel aux mêmes conditions qu'aux produits NT. Sybase SQL Server 11.0.3.3 est un produit déjà ancien et il lui manque certaines caractéristiques des produits les plus récents tels que Oracle 8i mais, compte tenu du prix et des performances, il a contribué au succès de beaucoup de projets antérieurs que j'ai menés. Comme je l'ai dit au début de ce chapitre, j'ai travaillé sur une base de données vraiment importante pour une société, sans qu'il soit question d'argent, et Sybase a parfaitement relevé le défi.

Cela dit, je dois une grande part de ce succès à l'excellence de l'assistance et de l'aide que j'ai obtenues de la plupart des acteurs en ligne cités dans ce chapitre. La liste de diffusion de Sybase sur Linux comprend beaucoup de bons DBA Sybase, de bonne volonté, et de développeurs qui répondent très volontiers aux questions, et qui, en tout cas, m'ont apporté des réponses.

© copyright Éditions Eyrolles

On peut souscrire à cette liste sur la page http://www.isug.com/ISUG2/ase_linux_form.html, et une archive est disponible sur http://forum.swarthmore.edu/epigone/ase-linux-list. Il y a aussi un groupe international d'utilisateurs Sybase (ISUG), qui propose beaucoup de services aux DBA et aux développeurs Sybase. Consulter leur site http://www.isug.com/.

FAQ

Q. Comment peut-on créer, dans Sybase, une colonne numérique qui s'incrémente automatiquement ?

R. Sybase se sert des colonnes IDENTITY à cette fin. La syntaxe pour créer une colonne identité est : **CREATE TABLE <tablename> (table_id NUMERIC(<size,>0) IDENTITY)**. D'habitude, ces colonnes servent à s'assurer qu'il existe une clé primaire sur la table : elles peuvent ainsi être spécifiées comme clés primaires. Voir, à la section « Problèmes », quelques-uns des problèmes qui touchent les colonnes identité.

Q. Je veux juste une petite base de données — est-ce que j'ai vraiment besoin de lui consacrer deux lecteurs physiques ?

R. Non. Il est recommandé, du point de vue de la restauration, que les journaux de transaction et les données soient situés sur des supports physique différents, pour le cas où l'un de ces supports physiques tomberait en panne. Si vous avez une petite base de données et que vous êtes prêt à prendre le risque d'une panne d'une unité, vous pouvez mettre les journaux de transaction et les données sur la même unité. En fait, vous pouvez spécifier que vos fichiers de base de données soient des fichiers du système des fichiers, si vous ne voulez pas leur consacrer une partition entière non formatée.

Q. Les index de mes tables ne semblent pas fonctionner. Que se passe-t-il ?

R. Sybase ne reconstruit pas les index à chaque fois que vous insérez de nouvelles données dans une table. L'optimiseur de requête de Sybase se voit impartir le travail, très dur, de décider si un index de la table est suffisamment à jour, ou si un balayage de la table peut s'avérer plus rapide. Si vous avez des tables dont les données changent, ou qui font l'objet d'ajouts fréquents, Sybase reconstruira périodiquement les index de ces tables. On peut forcer la mise à jour par la commande **update statistics <table>** dans **isql**.

Q. Mon journal de transaction remplit tout mon fichier de base de données, ce qui fait que les insertions échouent. Comment puis-je remédier à cela ?

© copyright Éditions Eyrolles

R. Sybase écrit une entrée dans le journal de transactions pour chaque transaction exécutée sur la base de données. En cas de modification importante des données par des **INSERT**, **UPDATE** ou **DELETE**, on peut en arriver à remplir le fichier journal de transaction, ce qui conduit la commande réelle à attendre, en arrière-plan, qu'il y ait assez d'espace dans le journal pour continuer. Pour forcer l'exécution de cette commande, se connecter à la base de données en tant qu'utilisateur **sa** et vider le journal de transactions pour la base de données en cause (spécifier l'option de mise à jour **no_log** si le journal est vraiment plein). Cela ne videra pas les transactions en cours d'exécution, si bien que si les changements se font en une seule grande transaction (par exemple, **DELETE FROM <table>** sur une grosse table) il se peut que cette action ne puisse pas se terminer parce qu'il n'y a pas assez d'espace dans le journal de transactions pour l'écrire tout entière. Dans ce cas, décomposer la transaction en tâches plus petites, les exécuter une à une, puis vider le journal de transactions.

Q. N'y a-t-il aucun moyen de vider automatiquement le journal de transactions ?

R. Si. Sybase offre la possibilité de tronquer le journal, chaque fois qu'il fait un checkpoint. Un checkpoint est une vérification périodique de Sybase qui sert à voir si les données du fichier des données sont cohérentes avec ce qu'elles devraient être. Il se produit toutes les deux secondes. On peut indiquer à Sybase de vider le journal de transactions à ce moment, en exécutant la commande suivante, connecté en tant que **sa** dans la base de données master :

```
sp_dboption <dbname>, trunc, true
```

© copyright Éditions Eyrolles

6

Installer DB2 version 6.1 sur Linux Red Hat

Dans ce chapitre

- Préparer l'installation de DB2 sur une station de travail Red Hat
- Installer DB2
- Configurer le centre de contrôle
- Installer un client DB2
- Configurer un client DB2 pour communiquer avec un serveur DB2

© copyright Éditions Eyrolles

Introduction

Dans un monde qui semble repousser de jour en jour ses frontières tout en se concentrant, il devient important de disposer de bases de données capables d'embrasser au mieux tout ce qu'il contient. Une base de données « universelle », pour mériter ce titre, se doit justement d'être : *universelle*. DB2 version 6.1 est vraiment une base de données universelle !

Voyons cela de plus près :

- **Universelle quant aux applications** de gestion de datawarehouse, d'aide à la décision, de recherche approfondie de données, de systèmes transactionnels (OLTP), et décisionnels (OLAP).
- **Universelle quant à sa capacité.** DB2 est la seule base de données sur le marché, capable de couvrir un domaine allant des applications de bureau aux systèmes de traitement parallèle haut de gamme, ce qui est vital pour des entreprises qui dépendent de transactions sur une grande échelle, de la maintenance de datawarehouse, et d'activités de recherche approfondie de données. Tout cela, allié à la fiabilité et à la puissance que l'on s'attend à trouver chez IBM, ainsi que la facilité de sa mise en œuvre — utilisation et installation —, a conduit *PCWeek* à baptiser DB2 « le serveur de base de données le plus avancé techniquement sur le marché ».
- **Universelle quant aux accès.** Dans le monde câblé d'aujourd'hui, une base de données se doit d'être universellement accessible, de sorte que tout utilisateur, quels que soient le système et le réseau, bénéficie d'accès faciles à la base de données. Non seulement DB2 fonctionne sur les systèmes d'exploitation les plus courants — Linux, Windows NT, Windows 9x, OS/2, AIX, PTX, HP-UX et Solaris Sun —, mais en plus on peut accéder à une base de données DB2 depuis une plage d'applications de client plus large. Mais ce sont surtout les supports Java et JDBC de DB2 qui rendent l'accès à la base de données vraiment universel. A l'aide de JDBC (Connectivité Java à la base de données), on peut fournir un accès client à DB2 *via* un Applet Java « léger ». Les applications de client léger JDBC fonctionneront avec n'importe quel navigateur Web équipé de Java : les personnes qui font une recherche d'informations peuvent ainsi accéder à une base de données, à partir d'une station de travail de leur organisme, en lançant un navigateur Web (Netscape Navigator fait partie de DB2).
  Il faut garder à l'esprit qu'aussi bien le cycle de développement, que la courbe d'apprentissage, pour des applications JDBC de client léger, sont très rapides : il est ainsi facile de se maintenir à niveau, malgré les changements qui interviennent dans les conditions de travail et d'échange, sans augmenter les coûts de programmation et d'apprentissage. Autrement dit, on peut toujours construire en Java des applications client hautement sophistiquées, tandis que l'on accède à des données DB2 par JDBC.

© copyright Éditions Eyrolles

- **Universelle quant à la gestion.** Au-delà la gestion des affaires sur le Web, de grandes masses d'information, des recherches complexes et du datawarehouse, DB2 facilite réellement la vie grâce à son outil de gestion de premier ordre, le centre de contrôle, qui simplifie la gestion de système, à la fois locale ou à distance, et, de ce fait, diminue le coût total du site.

Ce ne sont là que quelques-unes des caractéristiques perceptibles qui font de DB2 l'une des meilleures bases de données du monde. Mais on peut encore combiner la puissance de Linux avec la base de données la plus fiable au monde pour l'e-business, DB2 d'IBM. La base de données universelle DB2 pour Linux offre des fonctionnalités Internet sans égal. Le centre de contrôle DB2 propose aux administrateurs une interface prête à l'usage pour la maintenance de bases de données, qui peut être lancée depuis tout navigateur Web équipé de Java. Aux développeurs Java, DB2 pour Linux offre les supports JDBC et SQLJ, et Net.Data autorise la création d'applications Web dynamiques, pilotées par les données. Les développeurs Perl, quant à eux, peuvent tirer parti du driver Perl de DB2 pour l'interface de base de données Perl.

DB2 pour Linux est un très bon choix pour l'e-business. En tant que fondement de l'e-business, DB2 est l'un des systèmes de gestion de base de données relationnelles multimédia et « prêts pour le web » les mieux placés sur le marché, car il est assez fort pour répondre à la demande de grandes entreprises, et assez flexible pour se mettre au service des petites et moyennes entreprises. Avec la version 6.1, DB2 pour Linux se dote d'améliorations significatives qui facilitent le développement d'applications d'e-business. Voici quelques-unes des caractéristiques comprises dans le nouveau DB2 pour Linux :

- **Aide à l'amélioration de l'administration**. Contient le nouveau guide d'indexation et un moniteur de performance.
- **Améliorations du développement d'applications**. Comprend un support UNICODE amélioré, et des limites de table/colonne augmentées.
- **Support du container OLE DB**. Donne accès à des sources de données équipées de OLE-DB, telles que Microsoft Access et Lotus Approach.
- **Support de la connexion DB2**. Permet de protéger les sources des données et de maîtriser l'investissement courant qu'implique cette tâche, qu'il s'agisse de l'hôte ou de DB2 sur AS/400. Vu que DB2 gère plus de 70 pour cent des données informatiques du monde, le produit de connexion DB2 vous conduit à porter vos efforts sur la recherche et l'utilisation de vos données, plutôt que sur leur transmission.

Avec plus de 6500 entreprises maintenant associées au projet, le nombre des partenaires convertis aux solutions de gestion des données d'IBM a augmenté de 50 % en 1999. Plus de 50 000 programmeurs ont participé à la version 6.1 bêta de DB2 ; et maintenant on peut essayer la vraie!

© copyright Éditions Eyrolles

Ce chapitre vous fera parcourir les étapes à dérouler pour installer un serveur DB2, créer une base de données exemple, configurer les outils graphiques d'administration, installer un client DB2, et configurer une connexion entre un serveur DB2 et un client DB2.

Édition DB2 Personal Developer

Saviez-vous que l'on peut télécharger une version non limitée dans le temps de l'édition DB2 Personal Developer pour Linux ? Toutes les autres copies d'évaluation de DB2 sont limitées. L'édition DB2 Personal Developer fournit aux développeurs de logiciels tous les outils pour développer des outils de bureau et des applications, dans l'édition DB2 Universal Database Personal. Ce produit peut être téléchargé gratuitement, et utilisé pour l'évaluation, la démonstration, les tests et le développement de programmes d'application. Pour l'utiliser dans un environnement de production, il faut acquérir la licence appropriée, payante.

Sont comprises dans l'édition DB2 Personal Developer :

- l'édition DB2 Universal Database Personal, base de données DB2 toutes fonctions, mono-utilisateur. Cette version de DB2 peut se connecter à des bases de données DB2 à distance, mais ne peut répondre à des requêtes internes du client.
- l'édition DB2 Connect Personal pour connecter des applications de la station de travail aux bases de données hôte et AS/400. Avec ce produit, une station de travail accède aux données DB2 qui résident sur le système hôte ou AS/400.

Sont comprises dans les fichiers téléchargés, tant pour édition DB2 Universal Database Personal, que pour l'édition DB2 Connect Personal, les outils suivants :

- outils d'administration DB2, y compris le centre de contrôle.
- kit des développeurs de logiciel DB2, qui comprend une API, des outils et de la documentation destinée aux développeurs.

Les produits DB2 et l'édition DB2 Connect Personal sont disponibles pour Linux, OS/2, Windows 9x, et Windows NT.

Vous voulez en savoir plus ? Allez voir la page DB2 d'IBM pour Linux à http://www-4.ibm.com/software/data/db2/linux/[a].

a. Sur www-4.ibm.com/software/data/db2, la version disponible actuellement est la version 7.0.

© copyright Éditions Eyrolles

Préparer l'installation de DB2 sur une station de travail Red Hat

Avant d'installer un produit DB2 sur la station de travail Red Hat, il faut rendre cette dernière apte à recevoir DB2. Les tâches à effectuer pour cela dépendent de la version de Linux Red Hat. Comme le système d'exploitation Linux évolue, et que les relations se renforcent entre IBM et les éditeurs Linux, on peut s'attendre à voir disparaître ce type d'étapes de préparation. Jusque-là, pour que l'installation se déroule bien et que la configuration prenne le moins de temps possible, suivre soigneusement les instructions qui suivent.

Avant de pouvoir installer un produit DB2 sur une station de travail Linux, il faut s'assurer que la station de travail Linux répond aux contraintes suivantes :

- noyau Linux version 2.0.35 ou supérieure ;
- **RPM** (gestionnaire de paquetage Red Hat) ;
- paquetage **pdksh** (shell Korn du domaine public) ;
- **glibc** version 2.0.7 ou supérieure ;
- **libstdc++** version 2.8.0 ou supérieure.

Important. Si l'on utilise la commande **rpm –qa** pour lister les progiciels installés sur Red Hat version 6.0 ou version 6.1, seule apparaît la bibliothèque libstdc++ 2.9.0 ; au minimum, DB2 nécessite la bibliothèque libstdc++ 2.8.0, comprise dans l'installation de base, ce dont on s'assure en vérifiant le contenu du répertoire **/usr/lib**.

Préparer Red Hat version 5.2 et version 6.0 à une installation de DB2

Il est assez facile de réaliser une installation par défaut de Red Hat version 5.2, ou version 6.0, qui réponde à la question. Il suffit d'ajouter le paquet **pdksh** requis pour lancer programme d'installation de DB2, s'il n'est pas encore installé. Ce progiciel est disponible sur les CD-ROM Red Hat version 5.2 et version 6.0, dans le répertoire **/RedHat/RPMS**, et peut être installé à l'aide de la commande **rpm –i**.

Remarque. Si vous utilisez Gnome ou l'environnement KDE, vous pouvez utiliser un outil graphique pour ajouter ce progiciel.

Par exemple, pour installer le progiciel **pdksh-5.2.13-3** sur l'une de ces versions de Red Hat, entrer la commande suivante :

```
rpm –i pdksh-5.2.13-3
```

© copyright Éditions Eyrolles

On peut maintenant installer DB2 sur une station de travail qui fonctionne avec Red Hat version 5.2 ou version 6.0. Aller à la section *Installer la base de données universelle DB2 version 6.1 sur Linux Red Hat*, plus loin dans ce chapitre.

Préparer Red Hat version 6.1 à une installation de DB2

Depuis peu, la version 6.1 de Red Hat est, pour l'essentiel, disponible. Cette version de Red Hat présente un problème sur sa bibliothèque de chargement dynamique, qui provoque l'échec de DB2 au démarrage du gestionnaire de base de données pour une instance. Le gestionnaire de base de données, démarré par la commande **db2start**, est requis pour démarrer une instance, créer une base de données et accéder à des données dans une base de données. Bref, si le gestionnaire de base de données n'est pas démarré, on ne peut pas utiliser DB2. Comme c'est le programme d'installation de DB2 qui effectuera la plupart des étapes de configuration requises pour initialiser un serveur DB2, il lui faudra créer et démarrer une instance, ce qui aura pour effet de suspendre l'installation.

IBM travaille avec Red Hat à corriger ce problème ; en attendant, il y a quelques étapes à parcourir pour le contourner et réussir à installer DB2. Pour que la station de travail Red Hat version 6.1 installe un produit DB2, il faut télécharger un fichier de mise à jour (*patch*). Pour rendre les choses un peu plus confuses, la mise à jour à télécharger dépend de l'origine du code DB2 à installer. Votre code DB2 peut provenir de deux sources : la version de DB2 distribuée par IBM, disponible pour l'essentiel depuis fin 1999, ou la version de DB2 fournie dans la boîte de la version 6.1 de Red Hat.

Pour s'assurer que l'on télécharge le bon fichier de mise à jour et que l'implémentation est correcte, effectuer les étapes suivantes :

1. Se connecter à sa station de travail en tant qu'utilisateur avec l'autorisation *root*.

2. Se connecter au site ftp de support d'IBM à ftp://ftp.software.ibm.com /ps/products/db2/tools.

3. Télécharger le fichier de chemin approprié.

   Pour installer la copie de DB2 livrée dans la boîte Red Hat version 6.1, télécharger le fichier de mise à jour **db2rh61fix.tgz**. Pour installer toute autre copie de DB2, télécharger le fichier **db2rh61gafix.tgx**.

   S'assurer que la copie de DB2 que l'on installe *ne vient pas* de la boîte Red Hat version 6.1, si l'on télécharge le fichier **db2rh61gafix.tgx**. On ne peut pas partager ces corrections.

4. Décompresser (unzip et untar) le fichier de mise à jour en entrant la

© copyright Éditions Eyrolles

commande :

```
tar xvf patch_name
```

où **patch_name** est le nom du fichier de mise à jour approprié, qui a été téléchargé pour la copie de DB2.

5. Implémenter la mise à jour. Après décompression de la mise à jour, il reste trois fichiers dans le répertoire. L'un de ces fichiers, appelé **readme.txt**, donne des instructions complètes sur la manière d'implémenter les corrections.

   Après avoir résolu le problème associé à la bibliothèque de chargement dynamique de la version 6.1, il faut penser à ajouter le paquet manquant **pdksh** requis pour lancer le programme d'installation de DB2. Ce progiciel est disponible sur le CD-ROM Red Hat version 6.1, dans le répertoire **/RedHat/RPMS**, et s'installe à l'aide de la commande **rpm –i**.

Remarque. Si vous utilisez Gnome ou l'environnement KDE, vous pouvez utiliser un outil graphique pour ajouter ce progiciel.

Par exemple, pour installer **pdksh-5.2.14-1** sur Red Hat version 6.1, entrer la commande :

```
rpm –i pdksh-5.2.14-1
```

On peut maintenant installer un produit DB2 sur une station de travail qui fonctionne avec Red Hat version 6.1.

Installer DB2

Cette section vous fera découvrir l'initialisation et l'installation d'un serveur DB2, ainsi que la création d'une base de données d'exemple, qui sera utilisée pour vérifier l'installation. Avant d'exécuter les tâches présentées dans cette section, il faut être sûr que toutes les étapes nécessaires à l'installation de DB2 sur la station de travail se sont bien déroulées, comme exposé à la section précédente. Une fois l'installation terminée, on la vérifie localement, en se connectant à la base de données d'exemple que l'on a créée et en accédant aux données.

Les directives données dans cette section supposent que vous installiez et configuriez votre serveur DB2 en vous servant des options par défaut fournies avec l'outil d'installation du programme d'installation de DB2. Ce dernier est un programme d'installation à base de menus, qui vous assiste de bout en

© copyright Éditions Eyrolles

bout dans les tâches d'installation et de configuration de DB2, qui prennent beaucoup de temps. Les directives données dans cette section supposent aussi que vous ne mainteniez pas de versions précédentes d'un produit DB2 sur votre station de travail, ou d'utilisateurs par défaut (**db2inst1**, **db2fenc1**, et **db2as**). S'il existe l'un de ces utilisateurs sur votre système, enlevez-le à l'aide de la commande **userdel**. Je vous recommande de suivre les instructions de ce chapitre sur une station de travail vierge de toute installation de produit DB2.

Vous pouvez installer DB2 à l'aide de la commande **rpm** ; mais cela vous demandera encore quelques efforts, à la fin de l'installation, pour configurer votre serveur correctement. Je vous recommande d'utiliser le programme d'installation de DB2 pour installer et maintenir tous les produits DB2.

Exécuter l'installation

Maintenant que la station de travail Red Hat est équipée de DB2, on est prêt à installer la base de données universelle DB2 version 6.1. Les directives données dans cette section supposent que l'on installe l'édition du groupe de travail de la base de données universelle DB2 sur une station de travail équipée de DB2 Red Hat, vierge de toute copie de DB2.

Remarque. Il se produit quelquefois des problèmes d'affichage, quand on lance le programme d'installation de DB2. Si votre affichage rencontre ces problèmes, vous pouvez appuyer sur CTRL-L à tout moment, pour rafraîchir votre écran. Pour éviter la plupart des problèmes d'affichage, je vous recommande d'installer DB2 dans une session de console virtuelle, c'est-à-dire une fenêtre de terminal en dehors de l'interface graphique standard, installée sur la plupart des stations de travail Red Hat. Pour passer à une console virtuelle, appuyer sur CTRL-ALT-F1. Pour revenir à l'interface graphique, appuyer sur CTRL-ALT-F7.

Pour installer DB2, effectuer les étapes suivantes :

1. Se connecter à la station de travail Red Hat en tant qu'utilisateur avec l'autorisation *root*.

2. Insérer le CD-ROM de la base de données universelle DB2 version 6.1 dans l'unité de CD-ROM.

3. Même si l'on installe DB2 sur une station de travail qui fonctionne avec Red Hat version 6.1, et que l'on se sert de l'image créée par le fichier de mise à jour préalablement téléchargé, il y a encore besoin d'utiliser le CD-ROM DB2 : pour installer DB2, le fichier de mise à jour se sert des fichiers qu'il crée, en même temps que des fichiers trouvés sur le CD-ROM DB2. Pour plus d'information, voir le fichier **readme.txt** fourni avec la mise à jour.

© copyright Éditions Eyrolles

4. Monter le CD-ROM en entrant la commande :

```
mount –t iso9660 –o ro /dev/cdrom /mnt/cdrom
```

5. Se mettre dans le répertoire CD-ROM monté, en entrant la commande :

```
cd /mnt/cdrom
```

6. Si l'on installe DB2 sur une station de travail qui fonctionne avec Red Hat version 6.1, ne pas oublier de se mettre dans le répertoire créé à l'implémentation du fichier de mise à jour. Pour plus d'informations, se reporter au fichier **readme.txt** fourni avec la mise à jour.

7. Entrer la commande **./db2setup** pour démarrer le programme d'installation de DB2. Le fenêtre d'installation de DB2 V6 s'ouvre, comme à la figure 6.1.

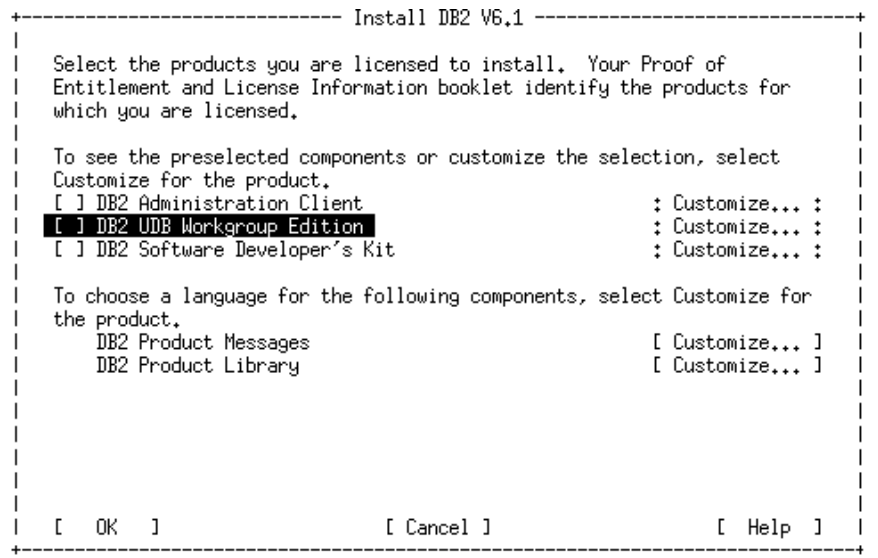
Install DB2 V6.1
Select the products you are licensed to install. Your Proof of Entitlement and License Information booklet identify the products for which you are licensed.
To see the preselected components or customize the selection, select Customize for the product.
[ ] DB2 Administration Client : Customize... :
[ ] DB2 UDB Workgroup Edition : Customize... :
[ ] DB2 Software Developer's Kit : Customize... :
To choose a language for the following components, select Customize for the product.
DB2 Product Messages [Customize...]
DB2 Product Library [Customize...]
[OK]
[Cancel]
[Help]

Figure 6–1
Fenêtre d'installation de DB2 version 6.1

8. Dans la liste des produits, choisir le produit DB2 à installer. Pour notre exemple, choisir **DB2 UDB Workgroup Edition**. Appuyer sur la touche TAB pour changer l'option en surbrillance, et sur la touche ENTRÉE pour choisir ou enlever la sélection d'une option. Pour plus d'informations, ou une assistance pendant l'installation de DB2, choisir **Help**.

9. Pour personnaliser l'installation, choisir **Customize,** à droite du produit DB2 sélectionné. Les options que l'on peut choisir pendant une installa-

© copyright Éditions Eyrolles

tion de DB2 sont montrées à la figure 6.2. **Select all** choisira tous les composants optionnels pour cette installation.

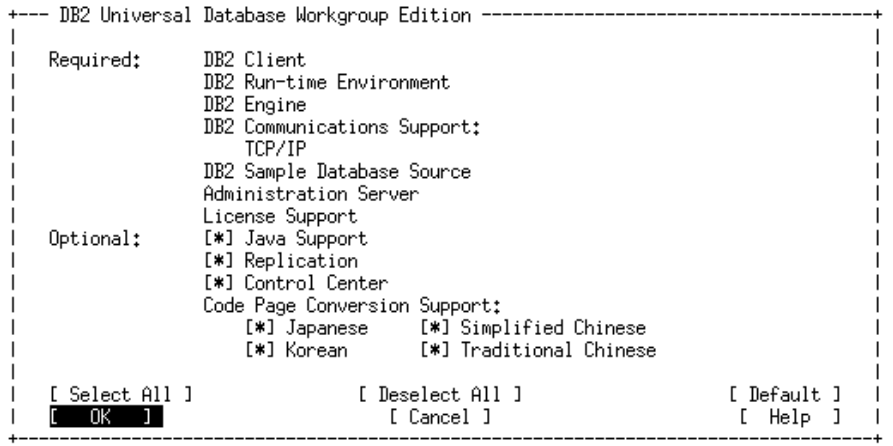

Figure 6–2
Fenêtre de personnalisation pour édition DB2 Workgroup

Dans notre exemple, cette liste peut être personnalisée, mais il faut s'assurer que l'on a au moins sélectionné les composants **Java Support** et **Control Center**, et choisir **OK**.

10. Pour personnaliser les messages de DB2, choisir l'option **Customize** appropriée. La fenêtre des messages de produit DB2 s'ouvre (figure 6.3). Les options par défaut sont bonnes. Choisir **OK**.

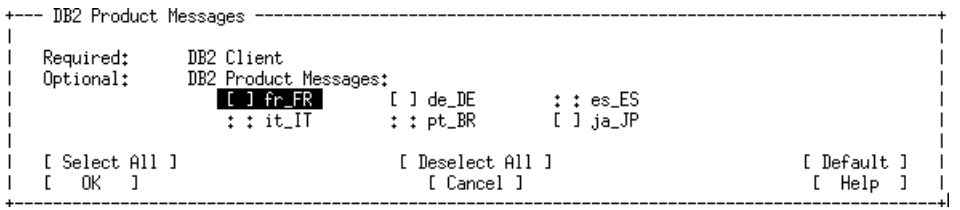

Figure 6–3
Personnaliser les messages de produit DB2

11. Pour personnaliser la bibliothèque de produit DB2, choisir l'option **Customize** appropriée. La fenêtre des messages de produit DB2 s'ouvre (figure 6.4). L'option par défaut sert à installer la documentation DB2 et l'aide en anglais pour le centre de contrôle. Pour notre exemple, les options par défaut sont bonnes. Choisir **OK**.

12. La fenêtre de création des services DB2 s'ouvre (figure 6.5).

© copyright Éditions Eyrolles

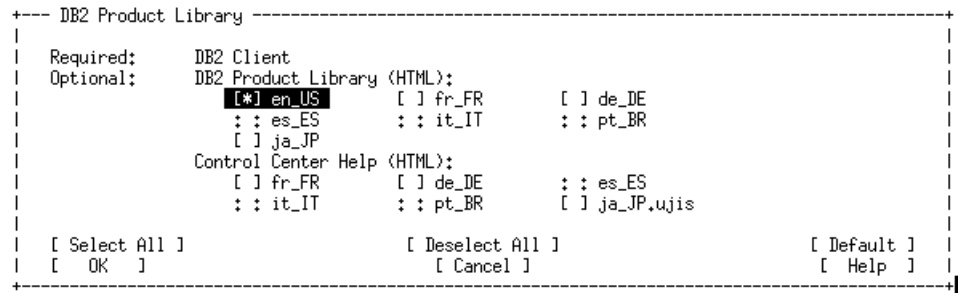

Figure 6–4
Personnaliser la bibliothèque de produit DB2

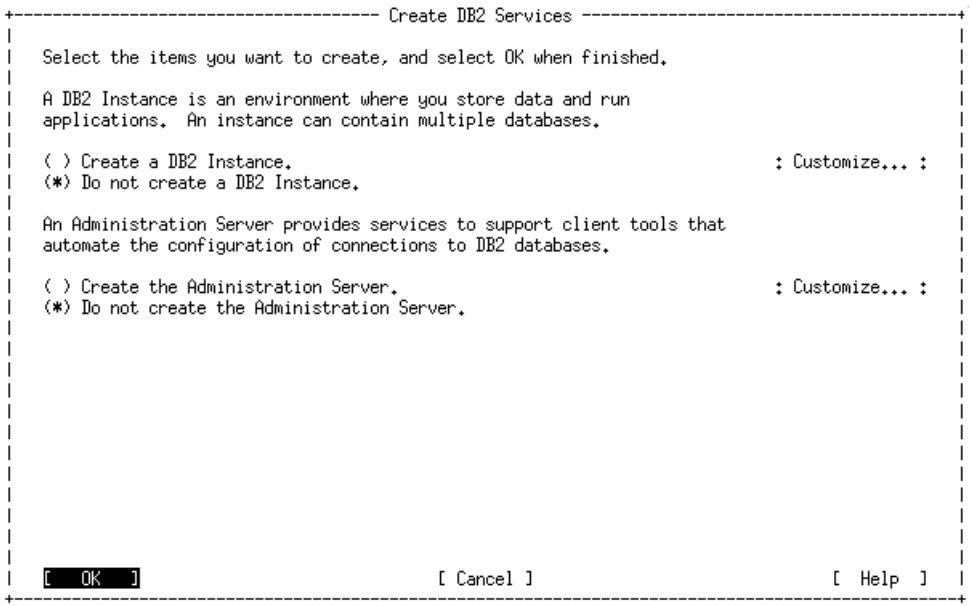
Create DB2 Services
Select the items you want to create, and select OK when finished.
A DB2 Instance is an environment where you store data and run
applications. An instance can contain multiple databases.
( ) Create a DB2 Instance.
: Customize... :
(*) Do not create a DB2 Instance.
An Administration Server provides services to support client tools that
automate the configuration of connections to DB2 databases.
( ) Create the Administration Server.
: Customize... :
(*) Do not create the Administration Server.
[OK]
[Cancel]
[Help]

Figure 6–5
Fenêtre de création des services DB2

13. Choisir **Create a DB2 instance** pour créer une instance, qui sera utilisée pour ranger la base de données d'exemple. Quand on choisit cette option, la fenêtre d'instance DB2 s'ouvre (figure 6.6).

    Le programme d'installation de DB2 générera un utilisateur par défaut, qui sera connu en tant que *propriétaire de l'instance*. Cet utilisateur aura l'autorisation administrateur système (SYSADM) sur l'instance. Un utilisateur qui a l'autorisation SYSADM sur une instance a le contrôle complet et l'autorité sur tous les objets de l'instance. Dans notre exemple, admettre les valeurs par défaut. Entrer un mot de passe dans le champ **Password** et

© copyright Éditions Eyrolles

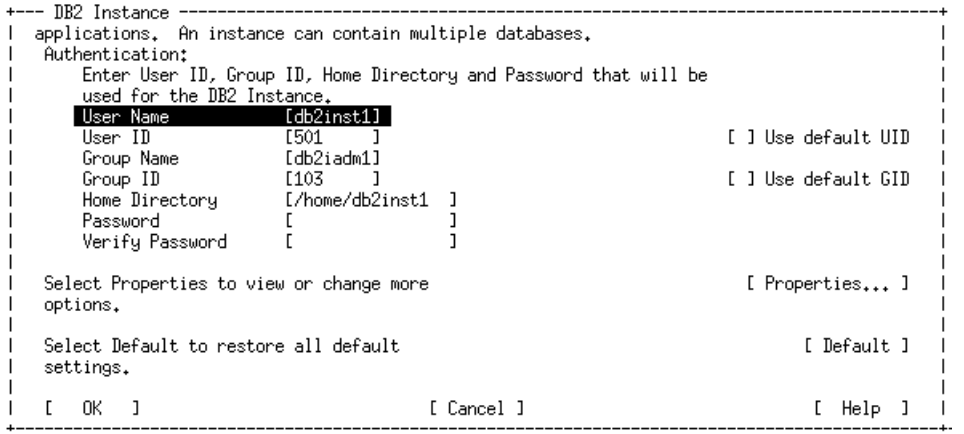
DB2 Instance
applications. An instance can contain multiple databases.
Authentication:
Enter User ID, Group ID, Home Directory and Password that will be used for the DB2 Instance.
User Name [db2inst1]
User ID [501] [ ] Use default UID
Group Name [db2iadm1]
Group ID [103] [ ] Use default GID
Home Directory [/home/db2inst1]
Password [ ]
Verify Password [ ]
Select Properties to view or change more options.
[Properties...]
Select Default to restore all default settings.
[Default]
[OK] [Cancel] [Help]

Figure 6–6
Fenêtre d'instance DB2

le vérifier dans le champ **Verify password**. Si l'on échoue à spécifier un mot de passe, le mot de passe par défaut **ibmdb2** sera utilisé. Je vous recommande de ne pas utiliser le mot de passe par défaut : comme il est utilisé pour toutes les installations de produits DB2, et est par conséquent bien connu, son utilisation présente un risque pour votre réseau.

Pour plus d'informations sur l'autorisation SYSADM, voir le *Guide d'administration de DB2*.

14. Choisir l'option **Properties**. La figure 6.7 illustre les propriétés que l'on peut spécifier.

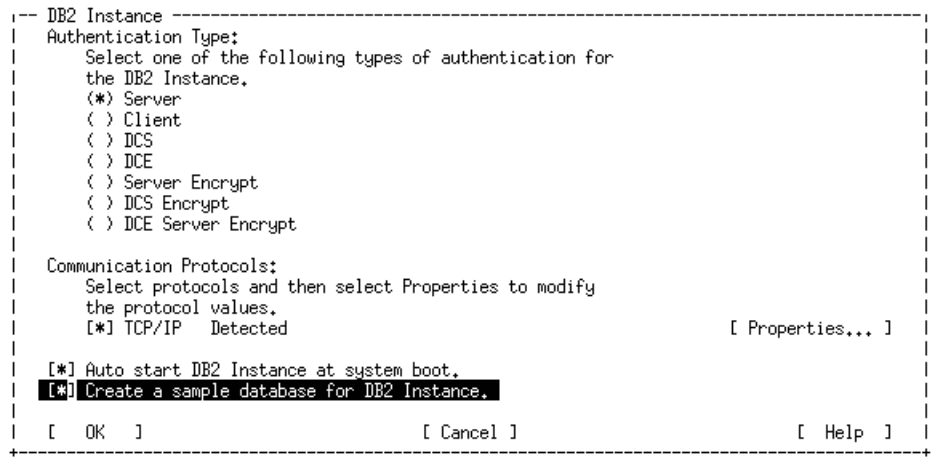
DB2 Instance
Authentication Type:
Select one of the following types of authentication for the DB2 Instance.
(*) Server
( ) Client
( ) DCS
( ) DCE
( ) Server Encrypt
( ) DCS Encrypt
( ) DCE Server Encrypt
Communication Protocols:
Select protocols and then select Properties to modify the protocol values.
[*] TCP/IP Detected
[Properties...]
[*] Auto start DB2 Instance at system boot.
[*] Create a sample database for DB2 Instance.
[OK] [Cancel] [Help]

Figure 6–7
Propriétés d'une instance

© copyright Éditions Eyrolles

Pour notre exemple, choisir les options **Auto start DB2 Instance at system boot** et **Create a sample database for DB2 Instance**. Le choix de ces options provoquera l'auto-démarrage de l'instance que le programme d'installation de DB2 crée, à chaque fois que le système sera relancé ; cela crée aussi une base de données exemple appelée **SAMPLE**. L'option d'auto-démarrage évite à l'utilisateur SYSADM d'avoir à démarrer l'instance à chaque redémarrage de la station de travail.

Les sélections par défaut pour le type d'authentification et les protocoles de communication sont bonnes dans notre exemple.

15. Choisir **OK.**

16. Vous serez invité à créer un compte utilisateur, qui servira à l'exécution des fonctions définies par l'utilisateur (UDF, ou User-Defined Functions) et des procédures stockées, comme on le voit à la figure 6.8.

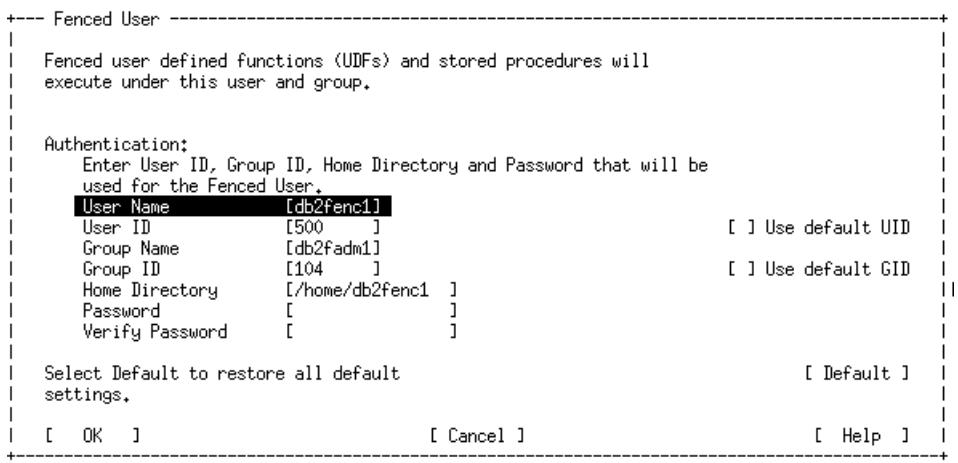
Fenced User
Fenced user defined functions (UDFs) and stored procedures will execute under this user and group.
Authentication:
Enter User ID, Group ID, Home Directory and Password that will be used for the Fenced User.
User Name [db2fenc1]
User ID [500] [ ] Use default UID
Group Name [db2fadm1]
Group ID [104] [ ] Use default GID
Home Directory [/home/db2fenc1]
Password [ ]
Verify Password [ ]
Select Default to restore all default settings.
[Default]
[OK] [Cancel] [Help]

Figure 6–8
Fenêtre d'utilisateur fenced

Dans notre exemple, admettre les valeurs par défaut, et spécifier un mot de passe différent pour cet utilisateur, comme pour l'utilisateur propriétaire de l'instance, puis choisir **OK** ; sachant que le mot de passe par défaut pour cet utilisateur est encore **ibmdb2**. On est renvoyé à l'écran de création des services DB2, comme on l'a vu à la figure 6.5.

Il n'est pas nécessaire de passer par cet utilisateur dans notre exemple. Pour plus d'informations sur les procédures stockées et les UDF, se reporter au *Guide d'administration*.

17. Choisir **Create the Administration Server** pour créer le serveur d'administration. Souvent appelé le DAS, il s'agit d'une instance spéciale, réservée

© copyright Éditions Eyrolles

aux tâches d'administration. Elle est utilisée par le centre de contrôle pour effectuer des tâches administratives à distance. Pour plus d'informations sur le serveur d'administration, se reporter au *Guide d'administration*. Quand on choisit cette option, la fenêtre du serveur d'administration s'ouvre (figure 6.9).

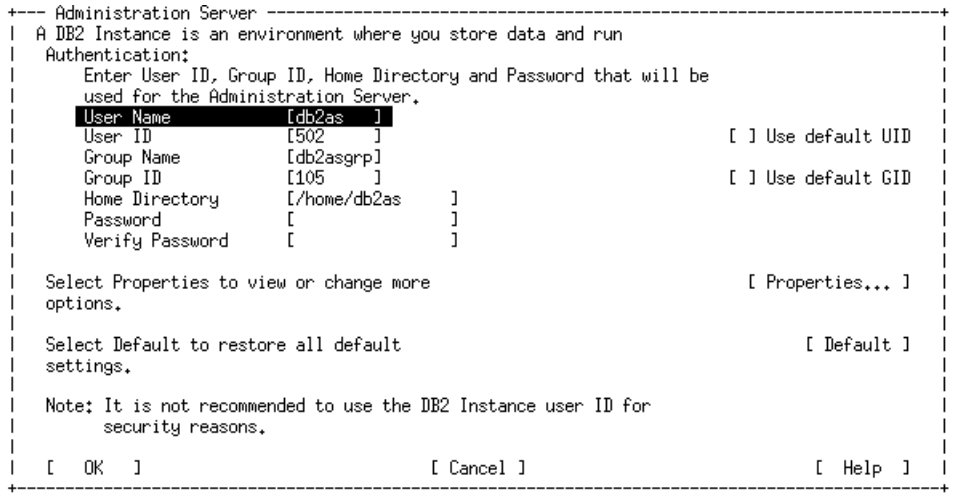
Administration Server
A DB2 Instance is an environment where you store data and run
Authentication:
Enter User ID, Group ID, Home Directory and Password that will be
used for the Administration Server.
User Name [db2as]
User ID [502] [ ] Use default UID
Group Name [db2asgrp]
Group ID [105] [ ] Use default GID
Home Directory [/home/db2as]
Password [ ]
Verify Password [ ]
Select Properties to view or change more options. [Properties...]
Select Default to restore all default settings. [Default]
Note: It is not recommended to use the DB2 Instance user ID for security reasons.
[OK] [Cancel] [Help]

Figure 6–9
Fenêtre du serveur d'administration

Le programme d'installation de DB2 générera un utilisateur par défaut, qui sera connu en tant que propriétaire de l'instance du serveur d'administration. Cet utilisateur aura l'autorisation administrateur système (SYSADM) sur l'instance du serveur d'administration. Pour notre exemple, admettre les valeurs par défaut. Entrer un mot de passe dans le champ **Password**, le vérifier dans le champ **Verify password**, puis choisir **OK**. Si l'on échoue à spécifier un mot de passe, c'est le mot de passe par défaut, **ibmdb2,** qui sera pris en compte. Je vous recommande de ne pas utiliser le mot de passe par défaut : comme il est utilisé pour toutes les installations de produits DB2, et est par conséquent bien connu, son utilisation présente un risque pour votre réseau. Une fenêtre contextuelle apparaît, qui vous donne votre nom de système DB2 (figure 6.10).

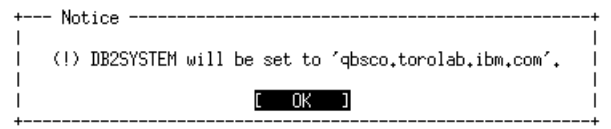

Figure 6–10
Fenêtre de nom DB2SYSTEM

© copyright Éditions Eyrolles

Noter ce nom, et appuyer sur **OK**. Retour à la fenêtre de création des services DB2, montrée à la figure 6.5.

18. Choisir **OK** pour commencer l'installation de DB2. La fenêtre de résumé s'ouvre (figure 6.11). Le résumé liste tous les composants que le programme d'installation de DB2 tentera d'installer et de configurer. Pour revoir la liste des composants, choisir **More** et utiliser les touches du curseur pour naviguer dans le rapport. Quand c'est terminé, choisir **Continue**.

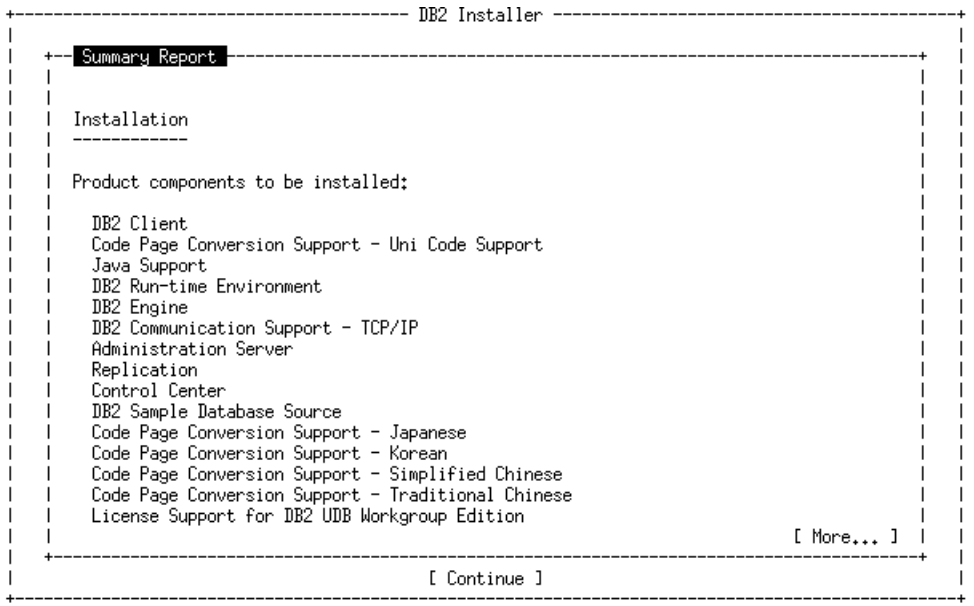

Figure 6–11
Fenêtre de résumé

Remarque. Si le résumé ne comprend pas les composants que vous voulez installer, ou s'il contient des composants que vous ne voulez *pas* installer, choisir encore **Continue**. Vous aurez la possibilité de revenir en arrière et de modifier ces sélections dans l'étape suivante.

19. Le choix **Continue** fait apparaître une fenêtre contextuelle, qui vous avertit que l'installation est sur le point de commencer, comme montré à la figure 6.12. Si vous êtes satisfait de vos sélections et voulez commencer à installer DB2, choisissez **OK**. Pour modifier vos sélections, choisissez **Cancel**.

© copyright Éditions Eyrolles

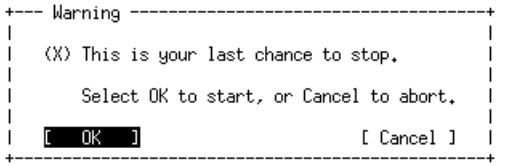

Figure 6–12
Fenêtre contextuelle d'avertissement

20. Quand l'installation est faite, la fenêtre de rapport d'état s'ouvre. Elle liste, comme à la figure 6.13, toutes les actions effectuées par le programme d'installation de DB2 et, pour chaque tâche, donne une indication sur son déroulement (succès ou non).

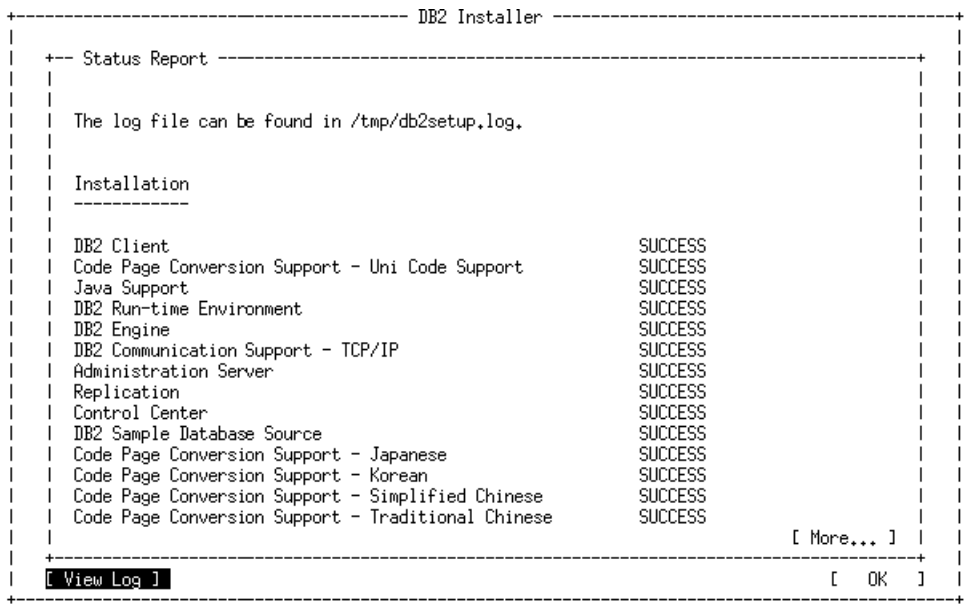
DB2 Installer
Status Report
The log file can be found in /tmp/db2setup.log.
Installation
DB2 Client SUCCESS
Code Page Conversion Support - Uni Code Support SUCCESS
Java Support SUCCESS
DB2 Run-time Environment SUCCESS
DB2 Engine SUCCESS
DB2 Communication Support - TCP/IP SUCCESS
Administration Server SUCCESS
Replication SUCCESS
Control Center SUCCESS
DB2 Sample Database Source SUCCESS
Code Page Conversion Support - Japanese SUCCESS
Code Page Conversion Support - Korean SUCCESS
Code Page Conversion Support - Simplified Chinese SUCCESS
Code Page Conversion Support - Traditional Chinese SUCCESS
[More...]
[View Log]
[OK]

Figure 6–13
Fenêtre de rapport d'état

Pour revoir la liste entière, balayer le contenu du rapport à l'aide des flèches. Si l'on reçoit des messages d'erreur immédiatement après l'installation, consulter la trace (log) de l'installation à l'aide de l'option **View Log**. La figure 6.14 présente un exemple de fichier log, qui contient les erreurs résultant des tâches d'installation.

Quand c'est terminé, choisir **OK**, puis **Close**, et désactiver toutes les fenêtres restantes en sélectionnant **OK.**

Vous avez fini d'installer votre serveur DB2 !

© copyright Éditions Eyrolles

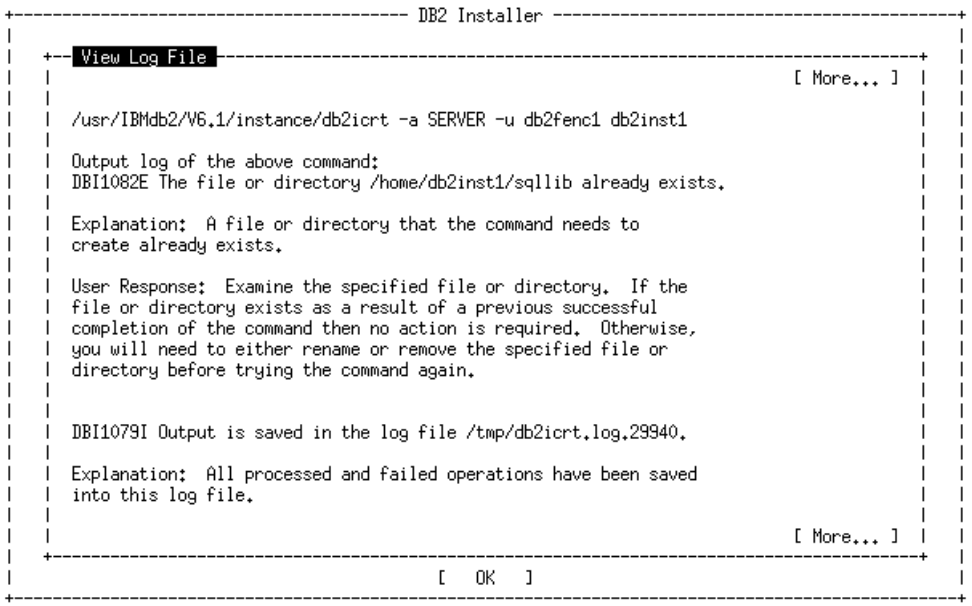
DB2 Installer
View Log File
[More...]
/usr/IBMdb2/V6.1/instance/db2icrt -a SERVER -u db2fenc1 db2inst1
Output log of the above command:
DBI1082E The file or directory /home/db2inst1/sqllib already exists.
Explanation: A file or directory that the command needs to create already exists.
User Response: Examine the specified file or directory. If the file or directory exists as a result of a previous successful completion of the command then no action is required. Otherwise, you will need to either rename or remove the specified file or directory before trying the command again.
DBI1079I Output is saved in the log file /tmp/db2icrt.log.29940.
Explanation: All processed and failed operations have been saved into this log file.
[More...]
[OK]

Figure 6–14
Le fichier log

Vérifier l'installation

Maintenant que DB2 est installé, accéder aux données de la base de données exemple (**SAMPLE**), pour vérifier l'installation.

Pour accéder à la base de données **SAMPLE**, parcourir les étapes suivantes.

1. Se connecter au système en tant que propriétaire de l'instance que le programme d'installation de DB2 a créée pendant l'installation. Dans notre exemple, entrer la commande :

```
su – db2inst1
```

Remarque. Si l'on utilise cette méthode pour se connecter au système, pour vérifier l'installation, il *faut* entrer cette commande avec le paramètre –, comme dans l'exemple précédent. Si l'on entre **su db2inst1**, on a en retour un message d'erreur, parce que DB2 ne lit pas correctement le profil DB2.

2. Démarrer le gestionnaire de base de données à l'aide de la commande **db2start**. Pour accéder à une base de données qui réside dans une instance, l'instance doit être démarrée et en fonctionnement. Souvenez-vous : nous

© copyright Éditions Eyrolles

avons opté pour cet auto-démarrage d'instance ; on reçoit le message suivant, après l'envoi de la commande :

```
SQL1026N The database manager is already active.
```

Remarque. Si vous recevez le message d'erreur **SQL10007N "-1393" ne peut pas être retrouvé. Code d'erreur : "1"**, n'ayez pas peur ! DB2 ne cherche tout simplement pas à connaître l'instance par défaut. Cela se produit, entre autres, si le fichier de profil DB2 n'a été pas exécuté à l'entrée dans le système (login). Dans cet exemple, entrer la commande : **export DB2INSTANCE=db2inst1**.

3. Entrer, pour se connecter à la base de données **SAMPLE,** la commande :

```
db2 connect to sample
```

On obtient la réponse suivante :

```
Database Connection Information

Database server = DB2/LINUX 6.1.0
SQL authorization ID = DB2INST1
Local database alias = SAMPLE
```

4. Entrer ensuite, pour choisir une liste de tous les employés qui appartiennent au département 20 dans la table **staff**, la commande :

```
db2 "select * from staff where dept = 20"
```

Important. Il faut entrer cette commande entre doubles quotes (" ") pour que Linux n'interprète pas l'astérisque comme le signe d'une commande de système d'exploitation. Quand on entre une commande DB2 comprenant un caractère qui peut avoir une signification spéciale au niveau du système, penser à mettre le contenu de la commande entre quotes, comme ici.

On obtient la sortie suivante :

```
  ID   NAME     DEPT  JOB    YEARS  SALARY    COMM
-----  -------- ----  -----  -----  --------  ---------
  10   Sanders  20    Mgr    7      18357.50  -
  20   Pernal   20    Sales  8      18171.25  612.45
  80   James    20    Clerk         13504.60  128.20
 190   Sneider  20    Clerk  8      14252.75  126.50

        4 record(s) selected.
```

© copyright Éditions Eyrolles

5. Entrer ensuite, pour restaurer la connexion à la base de données, la commande :

```
db2 connect reset
```

Félicitations ! Vous avez réussi à installer et configurer un serveur de l'édition Workgroup de DB2, et vous êtes prêt à vous servir de l'une des meilleures bases de données, sur la plate-forme la plus utilisée au monde pour l'e-commerce.

Pour utiliser le centre de contrôle, il faut d'abord le configurer : pour plus d'informations, voir la prochaine section. Si vous voulez installer un client DB2, voir la section *Installer un client DB2*, plus loin dans ce chapitre.

Configurer le centre de contrôle

Si l'on veut utiliser le centre de contrôle pour gérer des stations de travail à distance et planifier des scripts sur un serveur DB2, il faut initialiser l'environnement Run-Time Java (JRE). On peut lancer le centre de contrôle, soit comme une application Java, soit comme une applet Java au travers d'un serveur Web. Je vous recommande de lancer le centre de contrôle comme une application Java. Pour plus d'informations, se reporter au fichier README du centre de contrôle, appelé **readme.htm**, et situé dans le répertoire **/usr/IBMdb2/V6.1/cc/prime**. Les directives données dans cette section supposent que l'on initialise le centre de contrôle pour qu'il fonctionne comme une application Java. Si l'on ne souhaite pas utiliser le centre de contrôle, sauter cette section et aller directement à la section *Installer un client DB2*, plus loin dans ce chapitre.

Pour initialiser le centre de contrôle sur le serveur DB2, passer par les étapes suivantes :

1. Se connecter à la station de travail, en tant qu'utilisateur avec l'autorisation *root*.

2. Télécharger un JRE approprié. Pour avoir les informations les plus récentes sur les navigateurs et les JRE supportés, consulter

   http://www.software.ibm.com/data/db2/.

   Pour lancer le centre de contrôle, il faut au moins Java 1.1.7 v3, ou une version ultérieure.

   Un JRE pour Red Hat est disponible sur le site Web Blackdown à :

   http://www.blackdown.org/java-linux/mirrors.html.

© copyright Éditions Eyrolles

Pour cet exemple, j'ai exploré le site miroir à ftp://metalab.unc.edu /pub/linux/devel/lang/java/blackdown.org/JDK-1.1.7/i386/glibc/v3/, j'ai téléchargé le fichier **jre_1.1.7-v3-glibc-x86-native.tar.gz ;** puis je l'ai mis dans le répertoire **/tmp**.

Important. Il y a deux types de threads associés à un JRE : natifs et « verts ». Les threads affectent la manière dont Java fonctionne sur votre station de travail. Il vous appartient de télécharger la version native des threads du JRE que vous voulez utiliser. Le centre de contrôle DB2 ne supporte pas les threads verts.

3. Se mettre dans le répertoire où l'on a téléchargé le JRE. Dans notre exemple, ce sera le répertoire **/tmp**. Entrer la commande suivante pour atteindre ce répertoire :

```
cd /tmp
```

4. Une fois téléchargé le JRE approprié, il faut le décompresser (**unzip** et **untar)** en entrant la commande :

```
tar xvfz jre_1.1.7-v3-glibc-x86-native.tar.gz
```

5. Se connecter à la station de travail en tant que propriétaire d'instance. Dans notre exemple, se connecter à la station de travail en tant qu'utilisateur **db2inst1**.

Remarque. Vous pouvez utiliser, dans ce cas, la commande **su – db2inst1** ; mais assurez-vous que vous entrez cette commande avec le paramètre –, de sorte que le profil DB2 soit lu correctement.

6. Mettre à jour le **PATH**, pour que la station de travail connaisse l'emplacement des fichiers binaires du JRE installé. À supposer que vous ayez suivi l'exemple, entrez la commande :

```
# en shell Bash ou Bourne
        export PATH=/tmp/jre117_v3/bin:$PATH
# en shell C
        setenv PATH /tmp/jre117_v7/bin:${PATH}
```

où **/tmp/jre117_v7/bin** est le chemin des fichiers binaires du JRE que vous avez téléchargé.

Important. Cela ne crée le chemin que pour la session courante, ce qui suffit dans notre exemple. Pour le rendre permanent, il faut ajouter cette mise à jour à votre profil.

7. Démarrer le serveur d'applet JDBC du centre de contrôle en entrant :

```
db2jstrt 6790
```

© copyright Éditions Eyrolles

Le numéro de port 6790 est un nombre arbitraire ; tout numéro de port à quatre chiffres est valable, à condition qu'il ne soit pas en cours d'utilisation par une autre application.

8. La commande suivante démarre le centre de contrôle en se servant du port spécifié. Dans notre exemple, c'est le port 6790 :

```
db2cc 6790
```

Si l'on a spécifié un autre numéro de port au démarrage du serveur d'applet JDBC, c'est celui-là qu'il faut utiliser.

9. Cette commande ouvre la fenêtre d'identification du centre de contrôle (figure 6.15). Entrer un ID d'utilisateur et un mot de passe, défini sur la station de travail locale avec les autorisations correctes. Pour notre exemple, employer l'ID d'utilisateur **db2inst1** et le mot de passe spécifié pour cet utilisateur à l'installation du serveur DB2, puis cliquer sur **OK**.

Figure 6–15
Fenêtre d'identification du centre de contrôle

10. Le centre de contrôle apparaît (figure 6.16). On peut maintenant commencer à l'utiliser pour gérer les stations de travail DB2, locales et distantes. Pour plus d'informations, se reporter à l'aide en ligne du centre de contrôle ou au *Guide d'administration de DB2*.

Installer un client DB2

Cette section vous fera découvrir l'initialisation et l'installation d'un client DB2. Avant d'en parcourir les étapes, s'assurer que toutes les tâches nécessaires à l'installation d'un produit DB2 sur la station de travail se sont déroulées correctement, comme exposé à la section *Préparer l'installation de DB2 sur une station de travail Red Hat*, au début de ce chapitre.

© copyright Éditions Eyrolles

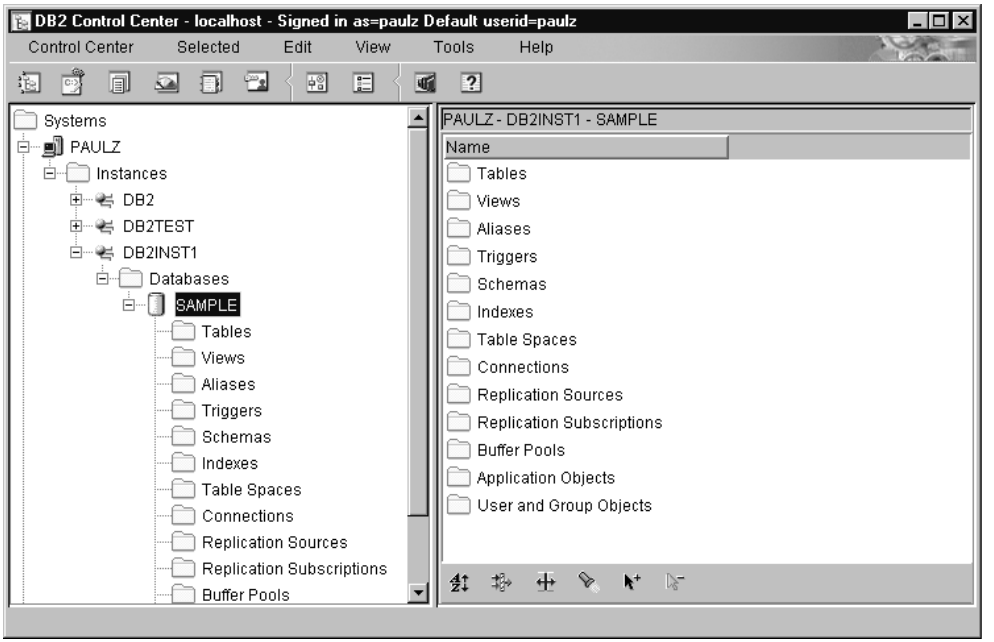

Figure 6–16
Le centre de contrôle

Les directives données dans cette section supposent que l'on installe et configure le client DB2 en se servant des options par défaut fournies avec l'outil d'installation du programme d'installation de DB2. Ce dernier est un programme à base de menus, qui vous assiste de bout en bout dans les tâches d'installation et de configuration de DB2, qui prennent beaucoup de temps. Les directives données dans cette section supposent aussi que vous ne mainteniez pas de versions précédentes d'un produit DB2 sur votre station de travail, ou l'utilisateur par défaut **db2inst.** Si vous avez cet utilisateur sur votre système, enlevez-le à l'aide de la commande **userdel.** Je vous recommande de suivre les instructions de ce chapitre sur une station de travail vierge de toute installation de produit DB2.

Important. Il se produit parfois des problèmes d'affichage, quand on lance le programme d'installation de DB2. Dans ce cas, appuyer sur la touche CTRL-L à n'importe quel moment, pour rafraîchir l'écran.

Pour éviter les problèmes d'affichage, installer DB2 dans une session de console virtuelle, c'est-à-dire une fenêtre de terminal hors de l'interface graphique standard, installée sur la plupart des stations de travail Red Hat. Pour passer à une console virtuelle, appuyer sur CTRL-ALT-F1. Pour revenir à l'interface graphique, appuyer sur CTRL-ALT-F7.

© copyright Éditions Eyrolles

Suivre les instructions de cette section pour installer un client Run-Time ou un client d'administration DB2. Dans notre exemple, nous installerons un client d'administration DB2, car il est disponible sur le CD-ROM de la base de données universelle DB2 version 6.1, préalablement utilisée pour installer le serveur DB2. Les clients d'administration DB2 sont aussi disponibles sur le CD-ROM *DB2 Administration Client* ; les clients Run-Time DB2 ne sont disponibles que sur le CD-ROM *Client DB2 Run-Time*. Pour plus d'informations, voir les *Suppléments d'installation et de configuration de DB2 et DB2 Connect*.

Pour installer un client DB2, parcourir les étapes suivantes :

1. Se connecter à la station de travail Red Hat en tant qu'utilisateur avec l'autorisation *root*.

2. Insérer le CD-ROM de la base de données universelle DB2 version 6.1 dans l'unité de CD-ROM.

   Même si l'on installe DB2 sur une station de travail qui fonctionne avec Red Hat version 6.1, et que l'on se serve de l'image créée par le fichier de mise à jour préalablement téléchargé, on a encore besoin du CD-ROM DB2 : le fichier de mise à jour se sert des fichiers qu'il crée, mais aussi des fichiers trouvés sur le CD-ROM DB2, pour installer DB2. Pour plus d'informations, voir le fichier **readme.txt** fourni avec la mise à jour.

3. Monter le CD-ROM en entrant la commande :

   ```
   mount –t iso9660 –o ro /dev/cdrom /mnt/cdrom
   ```

4. Se mettre dans le répertoire **cdrom** monté, en entrant la commande :

   ```
   cd /mnt/cdrom
   ```

   Si l'on installe DB2 sur une station de travail qui fonctionne avec Red Hat version 6.1, ne pas oublier de se mettre dans le répertoire créé à l'implémentation du fichier de mise à jour. Pour plus d'informations, se reporter au fichier **readme.txt**, fourni avec la mise à jour.

5. Entrer la commande **./db2setup** pour démarrer le programme d'installation de DB2. La fenêtre d'installation de DB2 V6 s'ouvre comme à la figure 6.17.

6. Dans la liste des produits, choisir le produit DB2 à installer. Dans notre exemple, choisir **DB2 Administration Client**. Appuyer sur la touche TAB pour changer l'option en surbrillance, et sur la touche ENTRÉE pour choisir ou enlever la sélection d'une option. Pour plus d'informations, ou une assistance pendant l'installation d'un client DB2, choisir **Help**.

© copyright Éditions Eyrolles

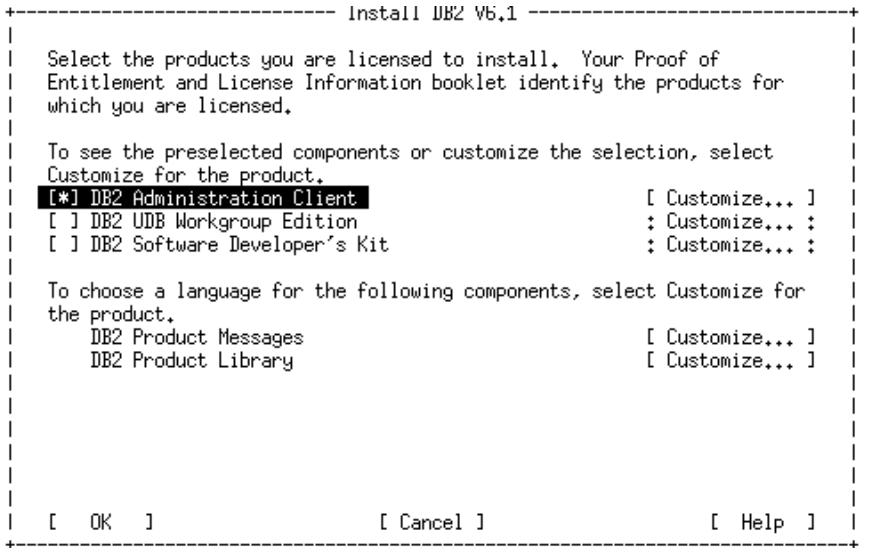
Install DB2 V6.1
Select the products you are licensed to install. Your Proof of Entitlement and License Information booklet identify the products for which you are licensed.
To see the preselected components or customize the selection, select Customize for the product.
[*] DB2 Administration Client [Customize...]
[ ] DB2 UDB Workgroup Edition : Customize... :
[ ] DB2 Software Developer's Kit : Customize... :
To choose a language for the following components, select Customize for the product.
DB2 Product Messages [Customize...]
DB2 Product Library [Customize...]
[OK] [Cancel] [Help]

Figure 6–17
Fenêtre d'installation de DB2 version 6.1

7. Pour personnaliser l'installation du client DB2, choisir **Customize** à droite du produit DB2 sélectionné. Les options d'installation du client d'administration DB2 sont montrées à la figure 6.18.

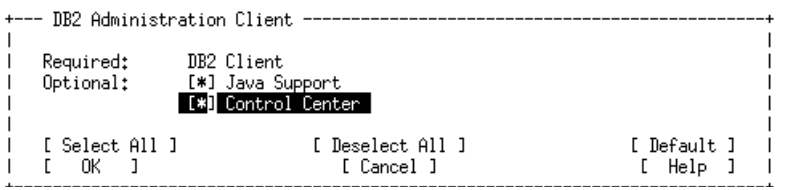
DB2 Administration Client
Required: DB2 Client
Optional: [*] Java Support
[*] Control Center
[Select All] [Deselect All] [Default]
[OK] [Cancel] [Help]

Figure 6–18
Fenêtre de personnalisation d'un client d'administration de DB2

8. **Select all** retiendra toutes les options pour cette installation de client DB2. Dans notre exemple, personnaliser cette liste de manière à installer les composants **Java Support** et **Control Center**, puis choisir **OK**.

9. Pour personnaliser les messages de DB2, choisir l'option **Customize** appropriée : la fenêtre des messages de produit DB2 apparaît (figure 6.19). Pour notre exemple, les options par défaut sont bonnes. Choisir **OK**.

10. Pour personnaliser la bibliothèque de produit DB2, choisir l'option **Customize** appropriée. La fenêtre des messages de produit DB2 s'ouvre (figure 6.20). La valeur par défaut ne signifie pas l'installation de la docu-

© copyright Éditions Eyrolles

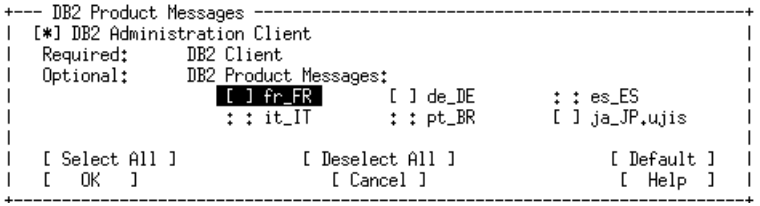

Figure 6–19
Personnaliser les messages de produit DB2

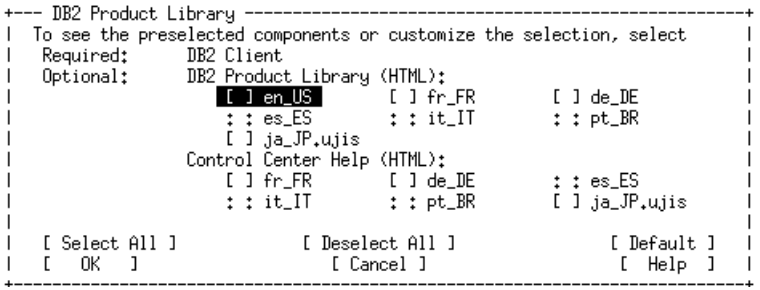

Figure 6–20
Personnaliser la bibliothèque de produit DB2

mentation DB2, mais l'aide du centre de contrôle, en anglais. Dans notre exemple, les options par défaut sont bonnes. Choisir **OK**.

11. La fenêtre de création des services DB2 s'ouvre (figure 6.21). Remarquez qu'elle est quelque peu différente de la fenêtre apparue quand vous avez installé un serveur DB2. Il manque le serveur d'administration DB2 : vous ne pouvez créer un serveur d'administration DB2 sur un client.

12. Choisir **Create a DB2 Instance** pour créer une instance, qui servira à stocker l'information requise pour se connecter à un serveur DB2 à distance, et à configurer l'environnement de la station de travail du client DB2. On ne peut pas ranger physiquement une base de données dans une instance du client, on ne peut que cataloguer l'information qui indique au client où est localisée la base de données à laquelle accéder. Quand on choisit cette option, la fenêtre d'instance DB2 s'ouvre (figure 6.22).

    Le programme d'installation de DB2 génère un utilisateur par défaut qui sera connu en tant que propriétaire de l'instance ; il aura l'autorisation administrateur système (SYSADM) sur l'instance. Un utilisateur qui a l'autorisation SYSADM sur une instance a le contrôle total et l'autorité sur tous les objets de l'instance. Dans notre exemple, admettre les valeurs par défaut. Entrer, dans le champ **Password,** le *même* mot de passe que celui de l'installation du serveur DB2, à vérifier dans le champ **Vérifier mot de passe**, et choisir **OK**.

© copyright Éditions Eyrolles

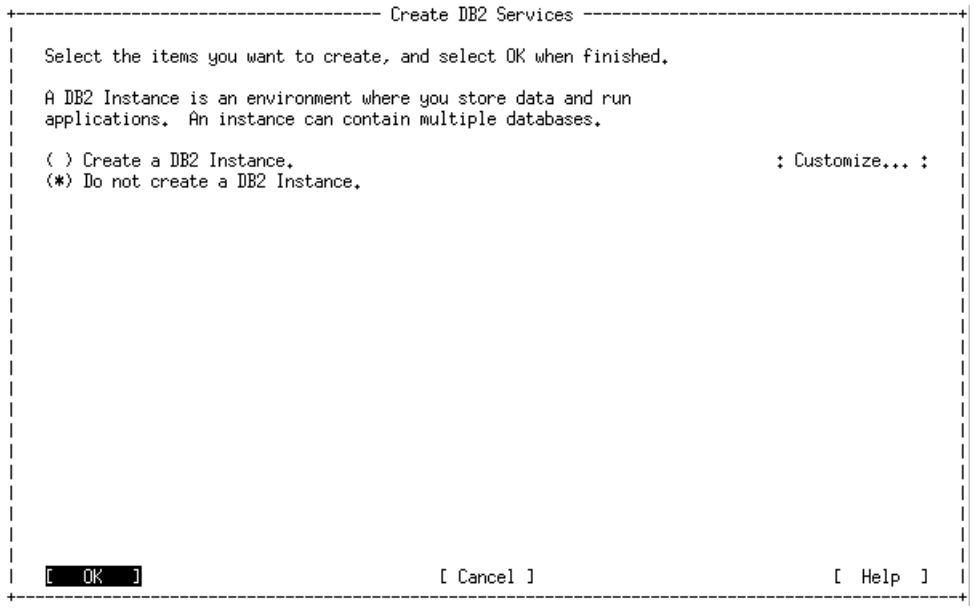
Create DB2 Services
Select the items you want to create, and select OK when finished.
A DB2 Instance is an environment where you store data and run
applications. An instance can contain multiple databases.
( ) Create a DB2 Instance.
(*) Do not create a DB2 Instance.
: Customize... :
[OK]
[Cancel]
[Help]

Figure 6–21
Fenêtre de création des services DB2 pour un client DB2

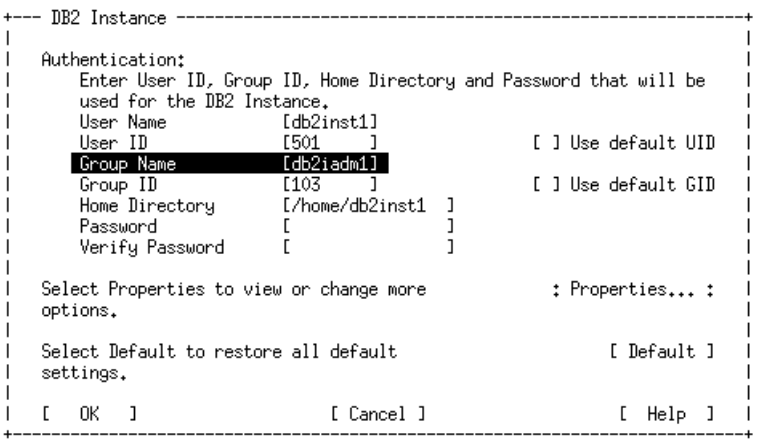
DB2 Instance
Authentication:
Enter User ID, Group ID, Home Directory and Password that will be
used for the DB2 Instance.
User Name [db2inst1]
User ID [501] [ ] Use default UID
Group Name [db2iadm1]
Group ID [103] [ ] Use default GID
Home Directory [/home/db2inst1]
Password [ ]
Verify Password [ ]
Select Properties to view or change more options.
: Properties... :
Select Default to restore all default settings.
[Default]
[OK]
[Cancel]
[Help]

Figure 6–22
Fenêtre d'instance DB2 pour un client DB2

En cas d'échec sur le mot de passe, c'est le mot de passe par défaut **ibmdb2** qui sera utilisé. Je vous recommande de ne pas utiliser le mot de passe par défaut : comme il est utilisé pour toutes les installations de produits DB2, et est par conséquent bien connu, son utilisation présente un

© copyright Éditions Eyrolles

risque pour votre réseau. Pour plus d'informations sur l'autorisation SYSADM, voir le *Guide d'administration de DB2.*

Remarque. Nous allons utiliser les mêmes ID utilisateur et mot de passe sur le serveur DB2 et sur le client DB2, pour simplifier notre scénario d'exemple.

13. Choisir **OK** pour commencer l'installation du client DB2. La fenêtre de résumé s'ouvre (figure 6.23). Le résumé liste tous les composants que le programme d'installation de DB2 tentera d'installer et de configurer. Pour revoir la liste des composants, choisir **More** et utiliser les flèches pour se déplacer dans le rapport. Après vérification, choisir **Continue**.

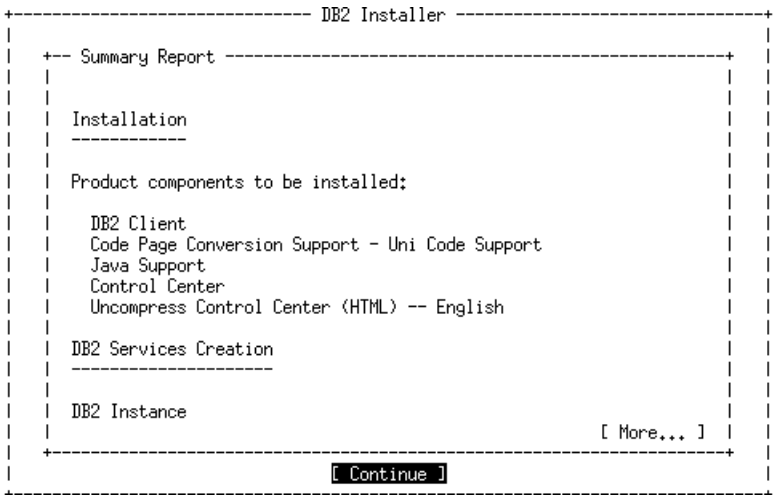

Figure 6–23
Fenêtre de résumé

Remarque. Si le résumé ne comprend pas les composants que vous voulez installer, ou s'il contient des composants que vous ne voulez *pas* installer, il faut quand même choisir **Continue**, pour revenir en arrière et éditer ces sélections à l'étape suivante.

14. Après que l'on a choisi **Continue**, une fenêtre contextuelle apparaît, pour signaler que l'installation va commencer (figure 6.24). Pour commencer à installer le client DB2, choisir **OK** ; pour modifier les sélections, **Cancel**.

15. Quand l'installation est terminée, la fenêtre de rapport d'état s'ouvre. Cette fenêtre (figure 6.25) liste toutes les actions effectuées par le programme d'installation de DB2, et indique, par tâche, le résultat. Pour revoir la liste entière, faire défiler le contenu du rapport à l'aide des flèches. En cas de messages d'erreur à la fin de l'installation, revoir la trace (log) de l'installation en sélectionnant **View Log**. Un exemple de fichier log est présenté à la figure 6.26.

© copyright Éditions Eyrolles

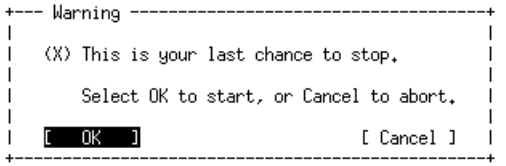

Figure 6–24
Fenêtre contextuelle d'avertissement

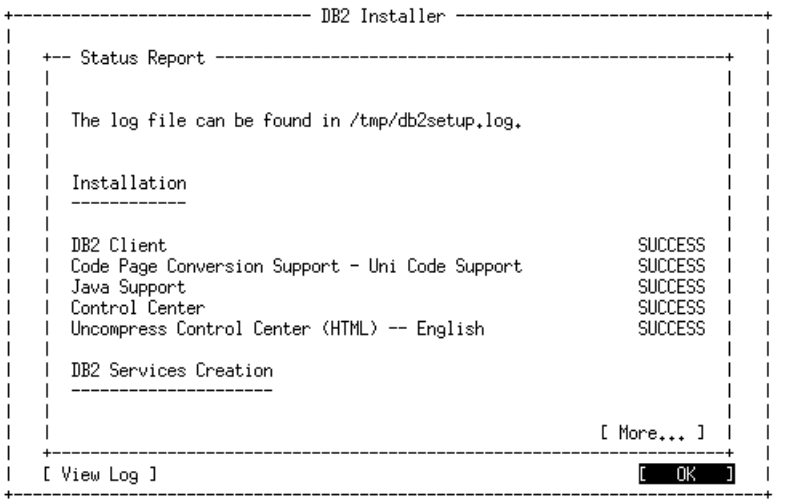

Figure 6–25
Fenêtre de rapport d'état

Enfin, choisir **OK**, puis **Close**, et désactiver toutes les fenêtres restantes en sélectionnant **OK.**

Félicitations ! Vous avez réussi à installer et configurer un client DB2, et vous êtes prêt à le configurer pour communiquer avec un serveur DB2 à distance. Attention : pour utiliser le centre de contrôle sur cette station de travail, il faut encore, auparavant, parcourir quelques étapes. Dans notre exemple, ce n'est pas nécessaire, car nous ne nous servirons pas du centre de contrôle sur la station de travail client de DB2. Si vous souhaitez utiliser ultérieurement le centre de contrôle sur votre station de travail, vérifiez les étapes franchies en vous reportant à la section *Configurer le centre de contrôle*, plus avant dans ce chapitre. Pour connecter un client DB2 à une base de données qui réside sur un serveur DB2 à distance, voir la prochaine section.

© copyright Éditions Eyrolles

DB2 Installer
View Log File
Log started at Tue Feb 1 15:39:25 EST 2000
The log file can be found in /tmp/db2setup.log.
Command to be executed:
/bin/rpm -ivh /mnt/cdrom/db2/db2cliv61-6.1.0-0.i386.rpm
Output log of the above command:
db2cliv61 ##
Command completed successfully.
[More...]
[OK]

Figure 6–26
Le fichier log

Configurer un client DB2 pour communiquer avec un serveur DB2

Cette section décrit comment configurer un client DB2 pour communiquer avec un serveur DB2, à l'aide du processeur de ligne de commande. On postule ici que TCP/IP fonctionne sur la station de travail du client ainsi que sur le serveur, et qu'il était en fonctionnement au moment de l'installation du serveur et du client DB2.

Avant qu'un client DB2 puisse être configuré en vue de communiquer avec un serveur DB2, le serveur doit être configuré de manière à pouvoir traiter des requêtes de client interne. Par défaut, le programme d'installation de DB2 configurera automatiquement le serveur DB2 pour les communications TCP/IP. Si l'on n'utilise pas le programme d'installation de DB2, il faudra configurer manuellement les communications du client interne à l'aide du processeur de ligne de commande. Pour plus d'informations sur la manière de configurer le serveur DB2 manuellement dans ce cas, se reporter au *Supplément à l'installation et à la configuration des connexions DB2*.

Remarque. Compte tenu des caractéristiques du protocole TCP/IP, celui-ci peut ne pas être immédiatement averti de l'échec d'une autre station de travail ; de sorte qu'une application cliente de DB2, accédant à un serveur DB2 à distance *via* TCP/IP, ou à l'agent équivalent sur le serveur, peut quelquefois sembler suspendue. DB2 se

© copyright Éditions Eyrolles

sert de l'option de socket TCP/IP SO_KEEPALIVE pour détecter s'il y a eu un échec, et si la connexion TCP/IP a été coupée. En cas de problème avec la connexion TCP/IP, se reporter au *Guide de dépannage DB2* pour plus d'informations sur le réglage de ce paramètre, et sur d'autres problèmes courants, relatifs à TCP/IP.

Pour configurer un client DB2 en vue de communiquer avec un serveur DB2 *via* TCP/IP, effectuer les étapes suivantes :

1. Identifier et enregistrer les paramètres TCP/IP requis. Ces paramètres sont présentés dans le tableau 6.1.

2. S'assurer que l'adresse TCP/IP du serveur DB2 à distance peut être résolue. Si l'on a l'intention de spécifier directement l'adresse IP du serveur DB2 à distance, ou que l'on dispose d'un serveur de nom sur le réseau, sauter cette étape. Cela se vérifie à l'aide de la commande **ping hostname**.

   Le client DB2 doit être capable de convertir l'adresse du serveur DB2 avec lequel il essaie d'établir la communication. S'il n'existe pas de serveur de nom sur votre réseau, vous devez spécifier directement un nom d'hôte qui corresponde à l'adresse IP du serveur dans le fichier local **/etc/hosts**.

Remarque. Si l'on veut maintenir la portabilité des clients DB2 de type UNIX, qui passent par les services d'information réseau (NIS), et que l'on ne dispose pas d'un serveur de nom, mettre à jour le fichier **/etc/hosts**, situé sur le serveur maître NIS.

   En tant qu'utilisateur doté de l'autorisation *root*, éditer le fichier local **/etc/hosts** du client et ajouter une entrée pour le nom d'hôte du serveur DB2. Par exemple :

```
9.21.125.124 db2server   # adresse d'hôte du serveur db2server
```

   où :

   – **9.21.125.124** est l'adresse IP du serveur DB2 à distance ;
   – **db2server** est le nom d'hôte du serveur DB2 à distance ;
   – **#** Signifie le début d'un commentaire qui décrit cette entrée.

Remarque. Si le serveur n'est pas dans le même domaine que le client, il faut donner un nom de domaine complètement qualifié tel que db2server.vnet.com, où vnet.com est le nom de domaine.

3. Mettre à jour le fichier local **/etc/services**. Si l'on a l'intention d'initialiser la communication entre un client et un serveur DB2, en n'utilisant que le numéro de port du serveur DB2 à distance, sauter cette étape.

   À l'aide d'un éditeur de texte local, et en tant qu'utilisateur doté de l'autorisation *root*, ajouter un nom de service arbitraire, ainsi que le numéro de port qui correspond au numéro de port du serveur DB2 à distance — con-

© copyright Éditions Eyrolles

Tableau 6–1
Paramètres TCP/IP requis pour la communication client-serveur DB2

Paramètre	Description	Exemple
Nom d'hôte (hostname) Ou Adresse IP (ip_address)	Utiliser le nom d'hôte ou l'adresse IP du serveur DB2 à distance. Pour renseigner ce paramètre : • Entrer la commande **hostname** sur le serveur DB2 pour obtenir le nom d'hôte. • Entrer la commande **ping hostname** pour obtenir l'adresse IP, ou contacter votre administrateur de système.	db2server Ou 9.21.125.124
Nom de service (svcename) Ou Numéro de port /Protocole (port_number/tcp)	Valeurs requises dans le fichier **/etc/services**. Le nom de service est un nom local arbitraire, qui représente le numéro de port de connexion sur un client DB2. Le numéro de port d'un client DB2 doit être le même que celui auquel correspond svcename dans le fichier **/etc/services** sur le serveur DB2. (Le paramètre svcename est défini dans le fichier de configuration du gestionnaire de base de données sur le serveur DB2). Ce numéro de port ne doit pas être en cours d'utilisation par d'autres applications, et doit être unique dans les fichiers **/etc/services**. Pour déterminer le paramètre svcename, se connecter au serveur DB2 en tant qu'utilisateur avec l'autorisation SYSADM et entrer la commande **db2 get dbm cfg**. Le paramètre svcename par défaut pour une installation de serveur DB2 par défaut est **db2cdb2inst1**, qui correspond au numéro de port par défaut **50000**. Si vous avez suivi les instructions de ce chapitre, vous pouvez utiliser ces valeurs par défaut.	**Db2cdb2inst1** correspond au port **50000/tcp**
Nom de nœud (node_name)	Un surnom local pour le serveur DB2 à distance auquel vous essayez de vous connecter. Un ordinateur est souvent considéré comme un nœud. Vous pouvez prendre n'importe quel nom ; mais toutes les valeurs de nom de nœud dans votre répertoire de nœud local doivent être uniques. Je vous recommande d'utiliser le nom d'hôte du serveur DB2 à distance, de manière à toujours savoir à quelle station de travail ce nœud se réfère.	db2server

© copyright Éditions Eyrolles

nexion au fichier des services client DB2 pour TCP/IP. Par exemple :

```
db2cdb2inst1 50000/tcp
# nom de service du client DB2 & port correspondant au serveur DB2
```

où :

- **db2cdb2inst1** est un nom local arbitraire, qui représente le numéro de port sur le client DB2.
- **50000** est le numéro de port auquel correspond le paramètre **svcename**, dans le fichier de configuration du gestionnaire de base de données sur le serveur DB2 distant — dans le fichier **/etc/services** du serveur DB2 distant. Le numéro de port utilisé sur le client DB2 doit correspondre au numéro de port utilisé sur le serveur DB2.
- **tcp** est le protocole de communication.
- **#** indique le début d'un commentaire qui décrit cette entrée.

Remarque. Si l'on veut maintenir la portabilité de clients DB2 de type UNIX, qui passent par les services d'information réseau (NIS), mettre à jour le fichier **/etc/services**, situé sur le serveur maître NIS.

4. Cataloguer un nœud TCP/IP. Ajouter une entrée au répertoire des nœuds du client DB2 pour décrire le serveur DB2 à distance. Elle spécifie le surnom choisi (**node_name**), le **hostname** ou l'adresse IP, et le **svcename** ou numéro de port que le client DB2 utilisera pour accéder au serveur DB2 distant.

   Pour cataloguer un nœud TCP/IP, effectuer les étapes suivantes :

   a. Se connecter au système en tant qu'utilisateur avec l'autorisation SYSADM ou SYSCTRL. Dans notre exemple, utiliser l'ID d'utilisateur du propriétaire d'instance, **db2inst1**.

Remarque. Si l'on utilise la commande **su** pour se connecter au système, il *faut* entrer cette commande avec le paramètre – par exemple, **su – db2inst1**. Si l'on n'utilise pas ce paramètre, on a en retour un message d'erreur, parce que DB2 ne lit pas correctement le profil DB2.

   b. Cataloguer le nœud TCP/IP en entrant la commande suivante, en mettant les variables paramètres entre crochets :

```
db2 "catalog tcpip node node_name  remote [hostname or ip_address]
server [svcename  or port_number]"
db2 terminate
```

   Par exemple, pour cataloguer le serveur à distance **db2server** sur le nœud appelé **db2server** avec, comme nom de service, **db2cdb2inst1**, entrer la commande :

© copyright Éditions Eyrolles

```
db2 "catalog tcpip node db2server remote db2server server db2cdb2inst1"
db2 terminate
```

Pour cataloguer un serveur à distance avec l'adresse IP **9.21.125.124** sur le nœud appelé **db2server**, en se servant du numéro de port **50000**, entrer la commande :

```
db2 "catalog tcpip node db2node remote 9.21.125.124 server 50000"
db2 terminate
```

5. Cataloguer la base de données. Avant qu'une application cliente puisse accéder à une base de données distante, cette dernière doit être cataloguée sur le serveur DB2, et sur tous les clients DB2 susceptibles de s'y connecter. Quand on crée une base de données sur un serveur DB2, elle est automatiquement cataloguée sur le serveur DB2 avec le même nom d'alias que son nom de base de données. Les informations contenues dans le répertoire de base de données servent, tout comme celles du répertoire de nœud, à établir une connexion à la base de données distante, sur le client DB2.

   Pour cataloguer une base de données, effectuer les étapes suivantes :

   a. Identifier et enregistrer les paramètres de base de données requis, présentés dans le tableau 6.2.

   b. Se connecter au système en tant qu'utilisateur avec les autorisations SYSADM ou SYSCTRL. Dans notre exemple, prendre l'ID utilisateur de propriétaire d'instance **db2inst1**.

   c. Cataloguer la base de données en entrant la commande :

```
db2 "catalog database database_name as database_alias at node
node_name"
db2 terminate
```

Remarque. Si l'on utilise la commande **su** pour se connecter au système, il *faut* entrer cette commande avec le paramètre – , par exemple **su – db2inst1**. Si l'on n'utilise pas ce paramètre, on a en retour un message d'erreur, parce que DB2 ne lit pas correctement le profil DB2.

Par exemple, pour cataloguer sur le nœud **db2server** une base de données distante, cataloguée avec l'alias de base de données **SAMPLE** (de sorte qu'elle ait localement l'alias de base de données **SAMPLE**), entrer la commande :

```
db2 "catalog database sample as sample at node db2server"
db2 terminate
```

© copyright Éditions Eyrolles

Important. Avant de pouvoir se connecter à la base de données distante, il faut avoir démarré le gestionnaire de base de données sur le serveur DB2 à distance. Dans notre exemple, l'instance créée était en auto-démarrage par défaut. Si vous avez interrompu cette instance, entrez, avant d'exécuter ces étapes, la commande **db2start**, en tant qu'utilisateur **db2inst1**, sur le serveur DB2 distant.

Tableau 6–2
Paramètres de base de données requis pour la communication client-serveur

Paramètre	Description	Exemple
Nom de base de données (database_name)	L'alias de base de données (**database_alias**) de la base de données DB2 à distance. Quand le programme d'installation de DB2 a créé la base de données **SAMPLE**, elle a été automatiquement cataloguée sur le Serveur DB2, avec le même nom en tant qu'alias de base de données et nom de base de données.	SAMPLE
Alias de base de données (database_alias)	Un surnom local pour la base de données distante, sur le client DB2. Si vous n'en donnez pas, ce paramètre prend par défaut la valeur du nom de base de données dans la table. Ce nom doit être unique sur le client DB2. Je vous recommande d'utiliser le même nom en tant que nom de base de données (l'alias de la base de données à distance) pour éviter les confusions. C'est ce nom qui est utilisé quand on se connecte à une base de données du client DB2.	SAMPLE
Nom de nœud (node_name)	Un surnom local pour le serveur DB2 distant auquel on veut se connecter. Un ordinateur est souvent considéré comme un *nœud*. Utiliser le même nom que quand vous avez catalogué le nœud dans l'étape précédente.	db2server

6. La configuration du client pour les communications est terminée ! Pour connecter le client DB2 au serveur DB2, effectuer les étapes suivantes :

   a. Se connecter au système en tant qu'utilisateur **db2inst1**.

   b. Entrer la commande suivante pour se connecter à la base de données distante :

```
db2 connect to database_alias user userid using password
```

   où :

   – **database_alias** est l'alias du client DB2 pour la base de données DB2 distante.

© copyright Éditions Eyrolles

– **userid** est un ID d'utilisateur défini sur le serveur DB2 distant. Dans notre exemple, utiliser **db2inst1**.

– **password** est le mot de passe pour le **userid** spécifié.

Si vous avez suivi l'exemple, et créé le même ID utilisateur, avec les mêmes mots de passe sur le serveur DB2 et sur le client DB2, vous n'avez pas à spécifier les paramètres **userid** et **password** dans cette commande. Par défaut, le client DB2 passera l'ID d'utilisateur courant et son mot de passe au serveur DB2, pour authentification. Comme ils sont identiques dans cet exemple, ce n'est pas d'un grand intérêt.

Pour se connecter, en tant qu'utilisateur **db2inst1,** à une base de données DB2 distante avec le nom d'alias de base de données **SAMPLE**, cataloguée sur le client DB2 avec le nom d'alias de base de données locale **SAMPLE**, entrer la commande :

```
db2 connect to sample
```

Important. On n'est pas obligé de donner les paramètres **userid** et **password** quand on se connecte à la base de données DB2 distante. A l'installation du client DB2, vous avez créé un utilisateur avec les mêmes ID d'utilisateur (**db2inst1**) et mot de passe que sur le client, pour simplifier le scénario d'exemple. Si vous aviez été connecté au client DB2 avec un ID utilisateur non encore défini sur le serveur DB2 distant, il vous faudrait utiliser les paramètres **userid** et **password** pour spécifier un utilisateur valide.

On obtient la sortie suivante :

```
Database Connection Information

Database server = DB2/LINUX 6.1.0
SQL authorization ID = DB2INST1
Local database alias = SAMPLE
```

c. Entrer la commande suivante pour sélectionner la liste de tous les employés du département 20 dans la table **staff** :

```
db2 "select * from staff where dept = 20"
```

Important. Il faut entrer cette commande entre doubles quotes (" ") pour que Linux n'interprète pas l'astérisque comme le signe d'une commande du système d'exploitation. Quand on entre une commande DB2 comprenant un caractère qui peut avoir une signification spéciale au niveau du système, penser à mettre le contenu de la commande entre quotes, comme ici.

© copyright Éditions Eyrolles

On obtient la sortie suivante :

```
 ID     NAME      DEPT      JOB       YEARS     SALARY    COMM
------  ---       -----     -----     ----      -------   ---------
 10     Sanders   20        Mgr       7         18357.50   -
 20     Pernal    20        Sales     8         18171.25  612.45
 80     James     20        Clerk               13504.60  128.20
190     Sneider   20        Clerk     8         14252.75  126.50

    4 record(s) selected.
```

d. Entrer la commande suivante pour réinitialiser la connexion à la base de données :

```
db2 connect reset
```

Félicitations ! Vous avez réussi à installer un client DB2 et à le connecter à un serveur DB2 à distance. C'était le but de ce modèle type de client-serveur, et vous en êtes venu à bout (pour installer et configurer, en tous cas).

Dépannage de la connexion

En cas de problèmes à la connexion du client DB2 au serveur DB2, vérifier les éléments suivants :

1. S'ils sont utilisés, les fichiers **/etc/services** et **/etc/hosts** ont été mis à jour correctement.

2. Le nœud a été catalogué avec les références correctes : nom d'hôte ou adresse IP, et nom de service ou numéro de port. Attention : le numéro de port (ou le nom de service) doit correspondre au numéro de port utilisé sur le serveur DB2. Pour vérifier ces initialisations, entrer la commande suivante, en tant qu'utilisateur **db2inst1** :

```
db2 list node directory
```

3. Le nom de nœud, spécifié quand on a catalogué la base de données, pointe l'entrée correcte dans le répertoire du nœud. On le vérifie en entrant la commande suivante, en tant qu'utilisateur **db2inst1** :

```
db2 list database directory
```

   S'assurer que le champ nom du nœud concorde avec le nom donné, à la création de l'entrée du nœud pour le serveur DB2 à distance.

4. La base de données sur le client DB2 est maintenant cataloguée correcte-

© copyright Éditions Eyrolles

ment, avec un nom d'alias de base de données du serveur DB2, catalogué à la création de la base de données sur le serveur DB2, en tant que base de données sur le client DB2. Par défaut, le nom d'alias de base de données du serveur DB2, pour un nom de base de données **SAMPLE**, est **SAMPLE**.

Si la connexion échoue encore après ces vérifications, voir le *Guide de dépannage*.

Résumé

Alors, où en sommes-nous ? Eh bien, croyez-le ou non, vous avez réussi là où nombre de sociétés de par le monde en sont encore à travailler dur! Installer un serveur DB2, le configurer pour les outils graphiques et les communications, puis créer une base de données exemple ; enfin installer un client DB2 et retrouver des données sur le serveur DB2 à travers une connexion LAN utilisant TCP/IP. Ce n'est pas rien !

Il n'est pas bien difficile de voir les progrès que Linux a accomplis en quelques années. Linux offre aux utilisateurs une stabilité, des fonctionnalités et une qualité qui lui permettent de se placer en bon concurrent des plates-formes actuellement sur le marché. Des millions d'utilisateurs du monde entier ont choisi Linux, pour des applications allant des serveurs Web et e-mail aux applications d'entreprise, verticales ou relatives aux services. Linux n'est plus désormais un projet d'amateurs, que les professionnels de l'informatique utilisent en coulisse, loin des regards indiscrets de la direction. Il a pris sa place, et ladite équipe de direction ferait bien d'en tenir compte. Il suffit de jeter un œil sur les projets des grands distributeurs, ou à ceux des constructeurs, qu'il s'agisse de logiciel ou de matériel, pour constater que Linux est présent. Voir aussi du côté de la bourse : impossible d'ignorer le récent démarrage en trombe des IPO de Linux, et l'engouement qu'ils déclenchent.

Oui, il y a vraiment une Linux-Mania, qui tombe en plein milieu de l'ère des bases de données. Maintenant vous avez les « billes » pour tirer parti de ce que d'aucuns considèrent comme parmi leurs meilleurs atouts, Red Hat et DB2.

© copyright Éditions Eyrolles

FAQ

Q. J'avais démarré le programme d'installation de DB2, mais j'ai dû en sortir anormalement. Quand j'essaie de le redémarrer, j'obtiens en retour le message d'erreur de la figure 6.27.

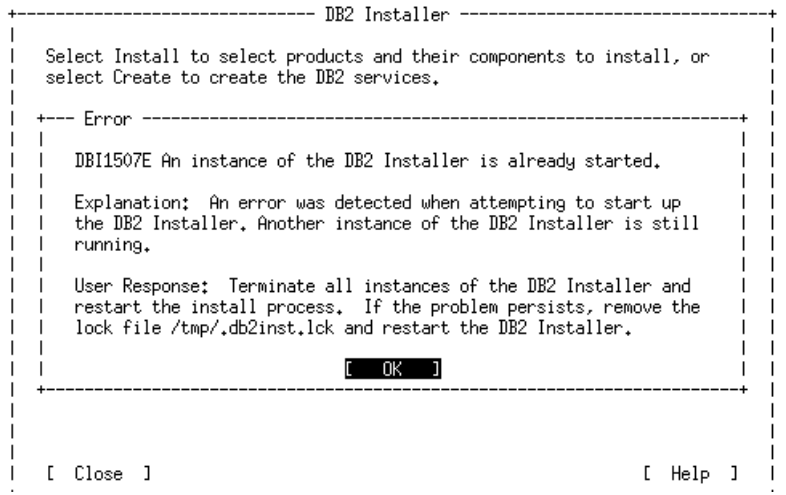
DB2 Installer
Select Install to select products and their components to install, or select Create to create the DB2 services.
Error
DBI1507E An instance of the DB2 Installer is already started.
Explanation: An error was detected when attempting to start up the DB2 Installer. Another instance of the DB2 Installer is still running.
User Response: Terminate all instances of the DB2 Installer and restart the install process. If the problem persists, remove the lock file /tmp/.db2inst.lck and restart the DB2 Installer.
[OK]
[Close]
[Help]

Figure 6–27
Erreur reçue quand une instance précédente de programme d'installation de DB2 ne s'est pas terminée correctement.

R. C'est parce que vous n'avez pas quitté le programme d'installation de DB2 correctement : le fichier **/tmp/.db2inst.lck** n'a été pas libéré. Pour redémarrer le programme d'installation de DB2, il vous faut l'enlever en entrant la commande :

```
rm /tmp/.db2inst.lck
```

Une fois ce fichier effacé, vous pouvez relancer le programme d'installation de DB2.

Q. Je n'ai *pas* fait le choix de suivre strictement vos instructions. À l'installation d'un serveur DB2, je n'ai pas choisi de créer le serveur d'administration. Ce faisant, j'ai reçu l'avertissement montré à la figure 6.28. Alors ?

R. C'est un avertissement standard qui est édité chaque fois que vous ne choisissez pas de créer le serveur d'administration. Je vous *recommande* de toujours choisir de créer le serveur d'administration. Sans le serveur d'administration, votre système ne sera pas capable d'administrer d'autres stations de travail DB2 à distance, de planifier les travaux locaux à lancer plusieurs fois, ou d'être disponible pour l'administration à distance.

© copyright Éditions Eyrolles

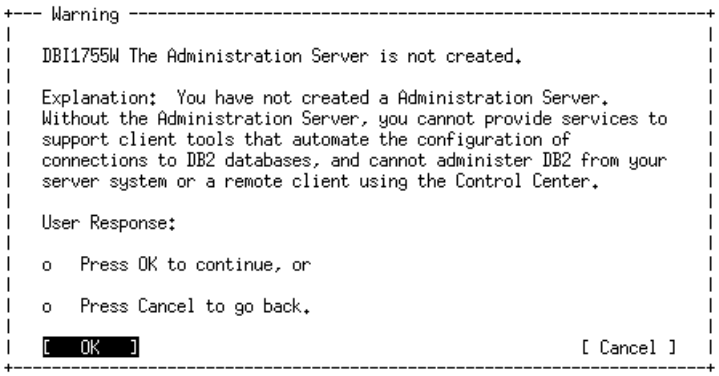
Warning
DBI1755W The Administration Server is not created.
Explanation: You have not created a Administration Server.
Without the Administration Server, you cannot provide services to
support client tools that automate the configuration of
connections to DB2 databases, and cannot administer DB2 from your
server system or a remote client using the Control Center.
User Response:
o Press OK to continue, or
o Press Cancel to go back.
[OK]
[Cancel]

Figure 6–28
Avertissement montré quand vous ne choisissez pas de créer le serveur d'administration sur un produit de serveur.

Q. Quand j'ai entré la commande **db2start** pour démarrer le gestionnaire de base de données de l'instance, j'ai reçu un message d'erreur : **SQL10007N "-1393" ne peut pas être retrouvé. Code erreur : « 1 »**.

R. DB2 ne se préoccupe pas de connaître l'instance par défaut. Quand vous entrez la commande **db2start**, DB2 tentera de démarrer l'instance par défaut. Cette dernière est déterminée par la variable d'environnement **DB2INSTANCE**, initialisée par le programme d'installation de DB2, à l'installation. Cette erreur se rencontre, le plus souvent, quand on se connecte au système avec la commande **su** sans le paramètre – , comme exposé précédemment dans ce chapitre. Si vous recevez ce message d'erreur, entrez la commande **exporter DB2INSTANCE=db2inst1** (bien sûr, si vous avez spécifié autre chose que l'instance **db2inst1** comme instance par défaut, c'est cette dernière qu'il faudra donner).

Q. J'ai utilisé les valeurs par défaut (**db2cdb2inst1** et **50000**) préconisées, quand j'ai configuré le client DB2 pour communiquer avec le serveur DB2. La connexion a échoué ! Je me suis reporté à la section Dépannage du client-serveur, et j'ai entré toutes les commandes correctement. Que s'est-il passé ?

R. Il peut y avoir beaucoup de raisons ; cela a probablement quelque chose à voir avec le **svcename** et le numéro de port. Aviez-vous encore une installation précédente de DB2 sur votre machine ? Les conseils, dans ce chapitre, supposent qu'il ne subsiste rien d'une éventuelle installation précédente de DB2 sur cette machine, et que le nom de service par défaut, ainsi que le numéro de port, soient disponibles. Si DB2 a déjà été installé sur cette machine, même s'il est supprimé, vérifier le fichier **/etc/services** pour voir si **db2cdb2inst1** correspond à **50000**. Peut-être, quand vous avez supprimé DB2, avez-vous oublié de nettoyer le fichier **/etc/services**? Dans

© copyright Éditions Eyrolles

ce cas, le nom de service, ou le numéro de port, n'est pas libéré pour le programme d'installation de DB2. Avez-vous changé l'une des options par défaut pendant l'installation ?

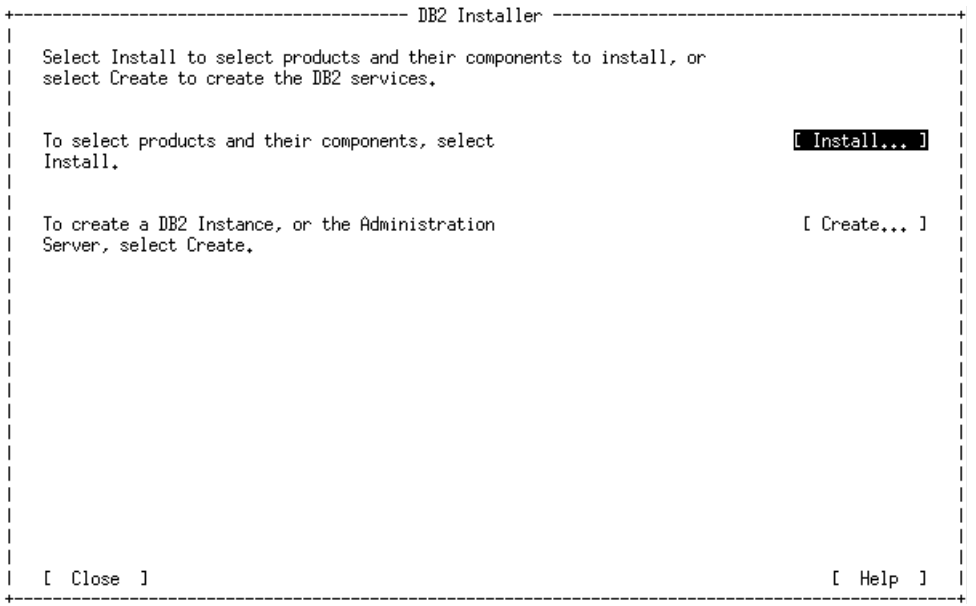
DB2 Installer
Select Install to select products and their components to install, or select Create to create the DB2 services.
To select products and their components, select Install.
[Install...]
To create a DB2 Instance, or the Administration Server, select Create.
[Create...]
[Close]
[Help]

Figure 6–29
Fenêtre du programme d'installation de DB2 après avoir installé un produit DB2.

Q. J'ai démarré le programme d'installation de DB2, après avoir installé un produit DB2. Pourquoi, tout à coup, est-ce la fenêtre de la figure 6.29 qui s'affiche ?

R. Après que l'on a installé un produit DB2, la fenêtre principale du programme d'installation de DB2 est toujours différente. C'est parce que l'on a davantage d'options. Vous pouvez installer un produit DB2 additionnel en sélectionnant **Install**, ou utiliser la caractéristique de création du programme d'installation de DB2 pour créer toutes les instances DB2 additionnelles, ou encore le serveur d'administration, en sélectionnant **Create**. Si vous utilisez le programme d'installation de DB2 pour créer une instance, ou le serveur **Administration**, ils seront configurés automatiquement pour les communications de client interne DB2 ; sinon, vous devrez les configurer manuellement. Je vous recommande d'utiliser le programme d'installation de DB2, pour créer, sur votre station de travail, les instances, ou le serveur d'administration.

© copyright Éditions Eyrolles

Q. Maintenant que j'ai un modèle de client-serveur DB2, où aller ?

R. Vaste question ! La réponse est : cela dépend de ce que vous voulez faire. Voici un itinéraire, ou plutôt une carte routière :

Tableau 6–3

Si vous voulez…	Allez voir…
Installer des clients DB2 multiples sans avoir à lancer le programme d'installation de DB2 à chaque station de travail client DB2	Supplément d'installation et de configuration de DB2 et DB2 Connect
En savoir plus à propos des fonctions de DB2, ainsi que la marche à suivre (par exemple, pour créer des bases de données, des tablespaces, des instances, des scénarios de sauvegarde et de restauration, etc.)	Guide d'administration de DB2
Charger des données dans vos bases de données ou exporter des données d'une base de données DB2 existante	Guide des utilitaires et des transferts de données avec DB2
Trouver plus d'informations sur ce que fait IBM avec Linux, DB2, ou d'autres produits IBM	http://www-4.ibm.com/software/data/db2/ ou http://www-4.ibm.com/software/is/mp/linux/
Trouver des réponses à des problèmes posés par DB2	Guide de dépannage de DB2
Accéder à des données qui résident sur un hôte ou une base de données DB2 AS/400	DB2 connexion rapide pour débutants
En savoir plus à propos du SQL de DB2	Guide d'introduction à SQL DB2 ou la référence SQL
Désinstaller DB2	Supplément d'installation et de configuration de DB2 et DB2 Connect ou aperçu rapide de DB2 pour UNIX

© copyright Éditions Eyrolles

7

Installer MySQL sur Linux

Dans ce chapitre

- Installation d'une distribution binaire ou RPM
- Installation d'une distribution source
- Configuration de la sécurité
- Optimisation des performances
- Problèmes et dépannage

© copyright Éditions Eyrolles

Introduction

MySQL est une base de données complexe, mais son installation ne l'est pas autant qu'on pourrait le penser. En préparant un peu les décisions avant de démarrer l'installation, on peut en faire un parcours paisible. Avant de commencer, il faut disposer d'une machine Linux et d'espace disque en abondance. La machine devra avoir à peu près de 30 à 40 Mo d'espace disponible pour les fichiers d'installation et la base de données. La mémoire est aussi un facteur clé — plus il y a de mémoire pour la base de données, mieux c'est. En général, 64 méga-octets suffiront.

MySQL est fourni dans des distributions variées, chacune présentant des aspects positifs et négatifs. Il y a des distributions binaires, des installations RPM (*Red Hat Package Manager* pour Red Hat Linux) et des distributions sources. Les distributions binaires et les distributions RPM sont les plus faciles à initialiser et peuvent s'exécuter en quelques minutes : elles initialisent et configurent la plus grande part de l'installation à votre place. Les distributions sources prennent plus de temps à installer et à initialiser, mais sont plus puissantes, et permettent à l'utilisateur de configurer et de régler son installation. Les distributions sources permettent de disposer librement d'une distribution binaire créée spécialement, qui peut être optimisée pour un type particulier de plate-forme, ce qui permet de générer une base de données qui fonctionnera aussi vite que possible sur une plate-forme donnée.

D'autre part, chaque distribution a son propre niveau de développement ; certaines sont des distributions à la limite des ressources, réservées à ceux qui veulent vérifier les nouvelles caractéristiques, même si elles tombent fréquemment en panne. Les versions stables, recommandées pour une première utilisation, contiennent toutes les caractéristiques majeures, et sont parfaitement robustes et bien testées. Les dernières versions sont destinées à ceux qui migrent depuis des versions anciennes et se placent derrière les versions stables. Elles ont seulement un rôle historique. Pour tout type de base de données appelé à être le support d'une application, il faut utiliser les versions stables. La plupart du temps, à moins d'avoir besoin d'une nouvelle caractéristique en cours de développement dans les nouvelles versions, une version stable fournira toutes les caractéristiques nécessaires.

L'installation de ces distributions, en particulier pour les distributions RPM et binaires, est traitée dans la suite du chapitre et ne prendra pas beaucoup de temps. Elles sont très faciles à installer et montrent la bonne voie pour tirer le meilleur parti de la puissance de MySQL dans une application. Une distribution source demande des étapes supplémentaires, mais pas très complexes et bien détaillées. Après avoir installé une distribution source, on peut s'attaquer à la plupart des autres distributions open-source. Pour avoir de la puissance, utiliser une distribution source qui optimise la vitesse et le fonctionnement de

© copyright Éditions Eyrolles

la base de données. Même si l'on n'installe pas les autres types de distributions, il est bon de les connaître et d'avoir une idée de la manière de les initialiser, pour le cas où l'on aurait à changer de plate-forme ou à migrer vers un autre type de distribution.

Une fois l'installation faite, les étapes suivantes concernent la sécurité de la base de données. S'assurer que seuls les utilisateurs autorisés peuvent accéder à la base de données est une étape très importante. Il faudra, selon les applications, être très attentif à la sécurité de la base de données, en particulier dans les applications d'e-commerce où l'on est en présence de données très sensibles, par exemple les numéros de carte de crédit. Chaque utilisateur devra avoir un mot de passe. Les utilisateurs auront des privilèges accordés, en fonction du type d'usage qu'ils font de la base de données. Si un utilisateur ne fait que consulter les tables, pourquoi lui donner le droit d'exécuter toutes les commandes sur la base de données ? On peut jouer des notions d'utilisateurs et de privilèges pour éviter les accès non autorisés à la base de données.

Toute base de données se doit d'être disponible en permanence. Créer des scripts personnalisés sur le serveur pour s'assurer que la base de données est toujours opérationnelle : les scripts de démarrage de la base de données s'assurent que la base de données fonctionne constamment et minimisent les temps d'arrêt. Il y a d'autres outils qui donnent des informations vitales à propos de la base de données et de son fonctionnement, qui facilitent l'administration de la base de données et permettent d'exécuter toutes les tâches courantes d'un administrateur. Il existe encore d'autres moyens d'optimiser les performances de l'installation MySQL, tels que les attributs de performance de la base de données MySQL, l'optimisation par recompilation de la distribution et les différentes options relatives au matériel. Toutes ces étapes mèneront à une application de base de données puissante.

Installation

Avant d'installer MySQL, il faut d'abord faire un choix parmi trois versions. Celle qui est recommandée est la version stable *3.22.X*. Elle est testée à fond, ne propose aucune caractéristique qui n'a pas été essayée, est prête pour la production et est aussi recommandée pour une première utilisation. La version de développement, *3.23.X*, est surtout destinée aux développeurs qui contribuent à l'amélioration de MySQL ; elle a beaucoup de caractéristiques nouvelles insuffisamment testées, et n'est pas recommandée dans un environnement de production. La version précédente, *3.21.X*, est une version désapprouvée, dont le principal intérêt est de donner, surtout aux anciens utilisateurs, un moyen d'accéder à la version 3.22 sans grand changement de

© copyright Éditions Eyrolles

fonctionnalité. Il s'agit plutôt d'une version historique que d'une version destinée à l'utilisation.

Après le choix de la version, il faut choisir entre distribution binaire, RPM et source. Une distribution binaire est déjà compilée : il suffit de l'installer et de choisir l'emplacement des fichiers. Le revers est qu'il faut trouver la distribution binaire appropriée à votre plate-forme, car on ne tient pas à jour les versions de toutes les plates-formes.

La distribution RPM est destinée à la distribution Red Hat Linux. Elle est similaire à une version binaire en ce sens qu'elle est déjà compilée et initialise l'environnement système. La dernière option est la distribution source. Cette distribution permet une meilleure personnalisation, est fournie avec beaucoup de documentation et d'exemples et sa vitesse peut être optimisée. Les versions binaires peuvent exiger des privilèges root mais les distributions RPM doivent obligatoirement être installées par le compte root. Binaires et RPM sont spécifiques de la plate-forme. La distribution source est très personnalisable et non spécifique d'une plate-forme ; elle peut être exécutée par tout utilisateur : c'est pourquoi elle est recommandée, de préférence aux autres formes de distributions.

Dans la suite de l'installation, il faudra localiser une distribution MySQL. Il est préférable d'obtenir la distribution depuis un site qui contient une copie des fichiers les plus récents du site officiel, c'est-à-dire un site miroir à proximité : cela diminuera probablement le temps de téléchargement. Les sites miroirs captent aussi une partie du trafic du site officiel, ce qui allège la charge de sa bande passante. La liste des sites miroir peut être consultée sur http://www.mysql.com/mirrors.html. Bien sûr, on peut toujours télécharger la distribution à partir du site officiel si l'on ne trouve pas de site miroir approprié. L'emplacement officiel est http://www.mysql.com/download.html.

Versions de MySQL, licences et assistance

MySQL est un paquetage *open-source*, ce qui signifie que le code source de la base de données et des outils associés est disponible gratuitement. Cela relève d'une conception puissante pour de nombreuses raisons. L'une d'elles est qu'en cas de problème avec la base de données, on est en mesure de le corriger, si l'on a des spécialistes à portée de la main. D'autre part, les logiciels *open-source* sont mis à jour en permanence : l'utilisateur dispose donc des caractéristiques les plus récentes et les plus importantes des bases de données, bien que cela complique parfois la gestion. Mais il ne faut pas trop en faire : chercher à tout prix à se maintenir à jour — qu'il s'agisse du logiciel ou des bibliothèques et outils associés — peut s'avérer quelque peu lourd.

© copyright Éditions Eyrolles

Les versions changent constamment, mais il n'est pas besoin de mettre à jour en permanence le système. Quand la distribution dans un environnement serveur est opérationnelle, on n'a pas besoin de la mettre à jour dès qu'une nouvelle version apparaît, surtout si les fonctionnalités nécessaires existent déjà dans votre application. Il vaut mieux attendre les dernières corrections apportées à la version majeure. Ajouter les nouvelles caractéristiques ; si une caractéristique est indispensable à une application, la mettre à jour immédiatement. De même, si l'on se sert fréquemment d'une fonction qui présente un bogue ou est cause d'un problème, la mettre à jour le plus vite possible pour alléger le problème.

MySQL n'exige aucune licence d'utilisation, même pour des applications commerciales. Si l'on vend le serveur équipé, et que cela comprenne la facturation de l'installation de MySQL, ou bien si MySQL fait partie d'une distribution payante, il faut une licence. On doit prendre une licence par processeur, mais les machines multiprocesseurs sont encore considérées comme une seule machine. Les prix pratiqués se trouvent sur la page http://www.mysql.com/Manual_chapter/manual_Licensing_and_Support.html#Cost. Au moment de la publication de cet ouvrage, le coût, pour un processeur, était de 200 dollars, avec des remises allant de 10 à 50 dollars.

Si vous utilisez MySQL pour des applications commerciales ou autres, il vous est proposé d'acheter une licence d'assistance qui contribuera au développement. Elle se présente sous quatre formes : e-mail de base, e-mail étendu, connexion et connexion étendue. Le tarif de base est de 170 euros, l'e-mail étendu de 1 000 euros, la connexion de 2 000 euros et la connexion étendue de 5 000 euros.

Si l'on veut utiliser une distribution binaire de MySQL, le mieux est de conserver la même plate-forme tout du long. Les distributions binaires ne sont pas mises à jour très fréquemment et les nouvelles versions peuvent rester indisponibles pendant un certain temps après parution de leur version de référence. Si vous utilisez une machine Linux sur la plate-forme Intel, vous aurez accès aux binaires les plus récents. Les autres plates-formes risquent d'être moins mises à jour que leur distribution binaire.

Conventions d'appellation

Les distributions MySQL sont dans le répertoire **Downloads**. Lors de la publication de cet ouvrage, la distribution de développement la plus récente était la version 3.22.32. Les exemples qui suivent montrent les différents types de formatage répertoriés pour chacune. Le X indique la version mineure.

© copyright Éditions Eyrolles

- Pour les distributions binaires, on trouvera ceci :
  - MySQL-3.22.X-PLATFORM.tar.gz
- Pour les distributions RPM :
  - MySQL-3.22.X-1.PLATFORM.rpm (le serveur)
  - MySQL-bench-3.22.X-1.PLATFORM.rpm (tests et benchmarks)
  - MySQL-client-3.22.X-1.PLATFORM.rpm (le client)
  - MySQL-devel-3.22.X-1.PLATFORM.rpm (client à personnaliser)
- Pour les distributions sources :
  - MySQL-3.22.X.tar.gz

Ce sont les fichiers dont vous avez besoin pour chaque distribution.

Remarquez que certains fichiers ont une étiquette supplémentaire : par exemple, *alpha*, *bêta* ou *gamma*. Elles indiquent le degré de maturité de la distribution. *Alpha* signifie qu'il y a de nouvelles caractéristiques, implémentées mais non testées : ce n'est pas une distribution au point et elle ne sert qu'à l'expérimentation. *Bêta* est une distribution avec de nouvelles caractéristiques qui ont subi un certain nombre de tests ; en général, des tests plus approfondis sont nécessaires, auxquels les utilisateurs peuvent contribuer. *Gamma* signifie une version stable destinée au plein usage. Encore une fois, sauf si avez vraiment besoin des nouvelles caractéristiques, les versions gamma sont les plus adéquates.

Pour décompresser les distributions, il faut quelques outils GNU, tels **tar** et **gunzip**, dont on vérifie la présence sur le système avec les commandes :

```
shell$ which tar
/bin/tar
shell$ which gunzip
/usr/bin/gunzip
```

Si un nom de chemin apparaît, c'est que l'outil est installé dans le répertoire indiqué. Si l'on ne dispose pas de ces outils, se reporter à la documentation de la distribution Linux pour savoir comment les ajouter. L'emplacement des fichiers est aussi déterminé par les options de la variable d'environnement PATH. Si rien n'apparaît, allez voir directement dans les répertoires. On peut aussi utiliser **rpm** pour Red Hat Linux :

```
shell$ rpm –q make
gzip-1.2.4-14
shell$ rpm –q gzip
make-3.77-6
```

© copyright Éditions Eyrolles

Ces outils s'installent à partir de la distribution Linux ou bien des distributions GNU. Pour plus d'informations, consulter le site web GNU : http://www.gnu.org/software/software.html.

Installation d'une distribution binaire

Après avoir téléchargé le fichier, le transférer dans le répertoire où l'on veut installer les fichiers, par exemple **/usr/local** (ce qui peut nécessiter le privilège root). Pour décompresser la distribution, exécuter la commande suivante dans le répertoire où l'on a transféré le fichier :

```
shell$ gunzip -c mysql-3.22.X-PLATFORM.tar.gz | tar -xvf –
```

Elle crée un répertoire du même nom que le fichier, **mysql-3.22.X-PLATFORM**. Créer un lien vers ce répertoire pour y faire référence plus facilement :

```
shell$ ln –s mysql-3.22.X-PLATFORM mysql
```

La prochaine étape installe et initialise la base de données et les tables associées. Se placer dans le nouveau répertoire **mysql** et exécuter la commande :

```
shell$ scripts/mysql_install_db
Creating db table
Creating host table
Creating user table
Creating func table
Creating tables_priv table
Creating columns_priv table

To start mysqld at boot time you have to copy support-files/mysql.server
to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
This is done with.:
/bin/mysqladmin -u root password 'new-password'
See the manual for more instructions.

Please report any problems with the. /bin/mysqlbug script!
The latest information about MySQL is available on the web at
http://www.mysql.com
Support MySQL by buying support/licenses at http://www.tcx.se/license.htmy.
```

La base de données peut être démarrée dans le même répertoire :

```
shell$ /bin/safe_mysqld &
```

et vous êtes maintenant prêt à configurer votre base de données.

© copyright Éditions Eyrolles

Installation d'une distribution RPM

Ce type d'installation nécessite le privilège root : ceci étant, elle exécute la configuration en entier et l'initialise à votre place. Il faut avoir téléchargé les fichiers RPM du client et du serveur indiqués auparavant, et ils doivent être dans la même version. Pour une installation minimale, sans client personnalisé, il suffit de :

- MySQL-3.22.X-1.PLATFORM.rpm
- MySQL-client-3.22.X-1.PLATFORM.rpm

Plus tard, il faudra aussi l'outil **rpm**, fourni avec la distribution Red Hat Linux. S'il n'a pas été installé, se reporter à la documentation Red Hat. Les fichiers de la base de données peuvent tous être installés simultanément comme suit :

```
shell# rpm -vi MySQL-3.22.X-1.PLATFORM.rpm MySQL-client-3.22.X-x.PLATFORM.rpm
```

Pour vérifier s'ils sont bien installés :

```
shell$ rpm -q MySQL
MySQL-3.22.X-1
```

Cette opération installe tous les fichiers nécessaires et démarre la base de données. Elle initialise aussi dans le répertoire **/etc** les scripts nécessaires pour démarrer la base de données en même temps que la machine. Cette distribution crée aussi un utilisateur **mysql** dans la base de données, dont elle se sert lors de l'installation. Il est propriétaire des fichiers, ce qui évite l'altération ou la falsification. La prochaine étape configure la base de données.

Installation d'une distribution source

Dans ce cas, il faut des outils GNU supplémentaires : **gcc** et **make**. Pour savoir s'ils sont disponibles sur le système, utiliser les commandes :

```
shell$ which gcc
/usr/bin/gcc
shell$ which make
/usr/bin/make
```

L'outil **gcc** doit être en version 2.81 ou supérieure. La commande qui suit affiche la version disponible :

```
shell$ gcc –v
Reading specs from /usr/lib/gcc-lib/i386-redhat-linux/egcs-2.91.66/specs
gcc version egcs-2.91.66 19990314/Linux (egcs-1.1.2 release)
```

© copyright Éditions Eyrolles

Si ce n'est pas la bonne version, se reporter à la documentation de la distribution Linux pour savoir comment la mettre à jour. Visiter aussi le site web GNU déjà indiqué pour obtenir de l'aide sur l'installation de ces outils.

Pour décompresser la distribution, exécuter la commande suivante dans le répertoire où l'on a transféré le fichier :

```
shell$ gunzip -c mysql-3.22.X.tar.gz | tar -xvf –
```

Elle crée le répertoire **mysql-3.22.X**. Se placer dans ce répertoire et exécuter la commande **configure** pour déterminer les informations d'environnement et préparer la compilation. On peut passer certaines options à **configure**, qui les range dans d'autres variables système et écrase les options par défaut (**/usr/local/mysql**) :

- **prefix=/path** (répertoire où sont installés les fichiers)
- **localstatedir=/path** (où sont installés les fichiers de la base de données)
- **with-charset=CHARSET** (la langue à installer)

On trouve de nombreuses autres options dans la documentation :

```
shell$. /configure --prefix=/my/home/
```

Maintenant, on compile tout à l'aide de **make** :

```
shell$. /make
```

et on installe tous les fichiers dans le répertoire indiqué dans **prefix** :

```
shell$. /make install
```

Se placer dans le répertoire où les fichiers ont été installés et initialiser les tables nécessaires :

```
shell$. /scripts/mysql_install_db
Creating db table
Creating host table
Creating user table
Creating func table
Creating tables_priv table
Creating columns_priv table
To start mysqld at boot time you have to copy support-files/mysql.server
to the right place for your system

PLEASE REMEMBER TO SET A PASSWORD FOR THE MySQL root USER !
This is done with:
/usr/local/mysql/bin/mysqladmin -u root password 'new-password'
See the manual for more instructions.
```

© copyright Éditions Eyrolles

```
Please report any problems with the /usr/local/mysql/bin/mysqlbug script!

The latest information about MySQL is available on the web at
http://www.mysql.com
Support MySQL by buying support/licenses at http://www.tcx.se/license.htmy.
```

On peut maintenant démarrer le serveur de base de données :

```
shell$ /my/home/mysql/bin/safe_mysqld &
```

et vérifier si la base de données fonctionne, comme ceci :

```
shell$ ps -aux | grep mysql | grep -v grep
mysql 1877 0.1 0.7 10636 1004 ttya2 SN 17:14 0:00 /usr/sbin/mysqld
mysql 1879 0.0 0.7 10636 1004 ttya2 SN 17:14 0:00 /usr/sbin/mysqld
mysql 1880 0.0 0.7 10636 1004 ttya2 SN 17:14 0:00 /usr/sbin/mysqld
```

Ces lignes montrent que la base de données fonctionne, et sous quel utilisateur et donnent des informations système à propos du processus. Si ces lignes n'apparaissent pas, cela signifie que la base de données n'est pas démarrée. Il faut alors vérifier les étapes précédentes et s'assurer que l'on peut démarrer la base de données. Dans la plupart des configurations, des opérations telles que l'ajout d'utilisateurs et de privilèges ne peuvent se faire que lorsque la base de données fonctionne.

Configuration

La disposition du répertoire **mysql** est initialisée de deux manières différentes, selon la distribution qui a été installée. Pour les distributions binaires, il présente cet aspect :

bin	répertoire principal et programmes de support
data	les bases de données elles-mêmes et les fichiers journaux
include	fichiers d'entêtes
lib	bibliothèques
scripts	scripts utilisés pour l'installation
share/mysql	messages d'erreur
sql-bench	benchmarks

et pour une distribution source, il se présente ainsi :

bin	répertoire principal et programmes de support
include/mysql	fichiers inclus
info	documentation
lib/mysql	bibliothèques

© copyright Éditions Eyrolles

libexec	le serveur lui-même
share/mysql	fichiers journaux d'erreur
sql-bench	benchmarks and séries de test
var	les fichiers de bases de données et les journaux

Sécurité

On consolide une application, si l'on restreint l'accès des utilisateurs à la base de données et qu'on leur laisse la possibilité de n'exécuter que certaines actions. C'est une étape importante sur laquelle il faut insister. Si la base de données est simplement destinée à l'évaluation du produit, ce n'est pas nécessaire. Par contre, tout service ou application qui utilisera cette base de données pour stocker des données sensibles devra être sécurisé.

Commençons par changer le mot de passe de l'utilisateur root dans la base de données, comme suit :

```
shell$ /my/path/mysql/bin/mysqladmin -u root password 'new-password'
```

Il faut aussi s'assurer que tout nouvel utilisateur créé dans la base de données a un mot de passe.

Un autre moyen facile d'initialiser MySQL de manière sûre est de l'exécuter comme son propre utilisateur. Exécuter le démon comme un nouvel utilisateur et lui attribuer toutes les permissions de fichier : cela évite l'altération par d'autres utilisateurs et le rend plus sûr. De même, ne pas exécuter le démon comme root : cela déjoue les tentatives des pirates qui s'attaquent au démon pour obtenir les privilèges root. Si vous avez installé le RPM Red Hat, cette partie de la configuration vous concerne. Si vous avez installé une distribution source ou binaire, il vous faut parcourir d'autres étapes.

On crée un utilisateur pour exécuter la base de données, **mysql** par exemple. On l'insère dans le fichier des mots de passe à l'aide de **adduser** :

```
shell$ /usr/sbin/adduser mysql
```

Sitôt que l'entrée de cet utilisateur est créée dans **/etc/passwd**, on l'édite pour pointer le répertoire correct de la base de données (**/my/path/mysql/var**) avec le shell approprié (par défaut, c'est le bon). L'utilisateur doit être propriétaire de tous les fichiers, ce qu'on obtient avec **chmod** :

```
shell$ chmod –R mysql /my/path/mysql
```

© copyright Éditions Eyrolles

On peut s'assurer que seul ce nouvel utilisateur a la permission d'écriture :

```
shell$ chmod –R go-w /my/path/mysql
```

MySQL propose aussi des options d'exécution pour le rendre plus sûr :

- **secure** recherche l'origine des adresses IP et s'assure qu'elles ne sont pas contrefaites. Cela ajoute une couche supplémentaire de protection contre les requêtes venant de l'extérieur, qui essaient de changer leur adresse à distance pour contourner les restrictions.
- **skip-networking** autorise seulement les connexions sur la machine locale, et non sur le réseau. C'est encore une bonne option si la machine sur laquelle fonctionne la base de données n'est pas protégée par un firewall.

Voir la documentation de MySQL pour plus de détails sur les améliorations de la sécurité et la protection du démon MySQL.

Privilèges

MySQL autorise les utilisateurs à exécuter certaines actions sur la base de données en fonction de leurs privilèges ; ceux-ci sont rangés dans une table et vérifiés quand l'utilisateur essaie d'exécuter une action. Le tableau suivant indique les privilèges accordés dans MySQL (tableau 7-1).

Tableau 7–1
Privilèges accordés dans MySQL

Privilège	Signification
select	lire une ligne dans une table
insert	créer une ligne dans une table
update	mettre à jour une ligne dans une table
delete	enlever une ligne d'une table
index	ajouter /supprimer un index sur une table
alter	autorise la commande SQL **ALTER TABLE**
create	créer une base de données ou une table
drop	supprimer une base de données ou une table
grant	donner des privilèges aux utilisateurs
file	lire ou écrire des fichiers externes à la base de données

© copyright Éditions Eyrolles

Tableau 7–1
Privilèges accordés dans MySQL

Privilège	Signification
reload	recharger ou vider des tables
shutdown	arrêter la base de données normalement
process	afficher ou supprimer des processus

En général, les utilisateurs ne disposeront pas des fonctions les plus puissantes, telles **grant**, **alter**, **file**, **process** et **shutdown**, réservées à l'administrateur.

Après avoir changé le mot de passe de root, ajoutons quelques nouveaux utilisateurs. Il y a essentiellement deux moyens d'ajouter de nouveaux utilisateurs à la base de données : soit on se sert de l'option **grant**, soit on insère, par des commandes SQL, de nouvelles lignes dans la table des utilisateurs. Avec **grant**, il faut d'abord s'assurer que la base de données est en route et utiliser **mysql** pour s'y connecter :

```
./mysql -p –user=root mysql
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or
Your MySQL connection id is 3 to server version: 3.22.30

Type 'help' for help.

mysql>
```

La table des utilisateurs ressemble à cela :

```
mysql> describe user ;
+-----------------+---------------+------+-----+---------+-------+
| Field           | Type          | Null | Key | Default | Extra |
+-----------------+---------------+------+-----+---------+-------+
| Host            | char(60)      |      | PRI |         |       |
| User            | char(16)      |      | PRI |         |       |
| Password        | char(16)      |      |     |         |       |
| Select_priv     | enum('N','Y') |      |     | N       |       |
| Insert_priv     | enum('N','Y') |      |     | N       |       |
| Update_priv     | enum('N','Y') |      |     | N       |       |
| Delete_priv     | enum('N','Y') |      |     | N       |       |
| Create_priv     | enum('N','Y') |      |     | N       |       |
| Drop_priv       | enum('N','Y') |      |     | N       |       |
```

© copyright Éditions Eyrolles

```
| Reload_priv     | enum('N','Y') |      |     | N       |       |
| Shutdown_priv   | enum('N','Y') |      |     | N       |       |
| Process_priv    | enum('N','Y') |      |     | N       |       |
| File_priv       | enum('N','Y') |      |     | N       |       |
| Grant_priv      | enum('N','Y') |      |     | N       |       |
| References_priv | enum('N','Y') |      |     | N       |       |
| Index_priv      | enum('N','Y') |      |     | N       |       |
| Alter_priv      | enum('N','Y') |      |     | N       |       |
+-----------------+---------------+------+-----+---------+-------+
17 rows in set (0.01 sec)
```

Le champ **host** contient l'emplacement depuis lequel l'utilisateur est connecté. La base de données connecte un utilisateur en fonction soit de son nom d'utilisateur et de son adresse IP, soit du nom d'hôte depuis lequel il se connecte. Ces informations, le nom d'hôte, le nom d'utilisateur et le mot de passe, sont comparées à la liste en vue d'accorder la connexion ou non. Le nom d'hôte peut être un nom complet, tel que **test.machine.com,** ou une adresse IP, comme **1.2.3.4**. Des caractères joker, symbolisés par des **%**, permettent d'autoriser une tranche plus large et d'être moins restrictif quand on stipule un domaine ou un réseau local tout entier : par exemple, **%.machine.com** ou **1.2.3.%**.

Dès que l'on voit l'invite **mysql**, cela signifie que l'on est au niveau de commande de la base de données et que l'on peut exécuter des commandes SQL. Voici, par exemple, les instructions **GRANT** pour créer un compte administrateur **dba** :

```
mysql> GRANT ALL PRIVILEGES ON *.* TO dba@localhost
       IDENTIFIED BY 'dbapassword' WITH GRANT OPTION;
mysql> GRANT ALL PRIVILEGES ON *.* TO dba@"%"
       IDENTIFIED BY 'dbapassword' WITH GRANT OPTION ;
```

ce qui autorisera le nouvel utilisateur **dba** à se connecter localement ou à distance de n'importe quel endroit, et à utiliser tous les privilèges de la base de données, le mot de passe de l'utilisateur **dba** étant **dbapassword**. Pour ajouter un utilisateur local disposant de moindres privilèges, on peut aussi utiliser la commande **GRANT** :

```
Mysql> GRANT SELECT,INSERT ON *.* TO testuser@localhost
       IDENTIFIED BY 'testuserpassword'
```

Cet utilisateur peut maintenant se connecter depuis la machine locale, avec le mot de passe **testuserpassword**, et exécuter des instructions **SELECT** et **INSERT** sur les tables. Si l'on connaît bien SQL, on peut contourner **GRANT** et insérer soi-même les données dans la table des utilisateurs, ainsi que manipuler cette table directement avec la commande **INSERT**.

© copyright Éditions Eyrolles

Voici comment créer le compte **dba** de l'exemple précédent :

```
mysql> INSERT INTO user VALUES('localhost','dba',PASSWORD('dbapassword'),
            'Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y')
mysql> INSERT INTO user VALUES('%,»dba,'PASSWORD('dbapassword'),
            'Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y','Y')
mysql> INSERT INTO user VALUES('localhost','testuser',
            PASSWORD ('testuserpassword'),
            'Y','Y','N','N','N','N','N','N','N','N','N','N','N','N')
mysql> FLUSH PRIVILEGES ;
```

Il existe deux autres tables pour accorder les droits d'accès. La table **db** tient la liste des utilisateurs qui peuvent accéder aux bases de données, des hôtes depuis lesquels ils se connectent et des privilèges qui leur sont accordés. Elle se présente comme ceci :

```
mysql> describe db ;
+-----------------+---------------+------+-----+---------+-------+
| Field           | Type          | Null | Key | Default | Extra |
+-----------------+---------------+------+-----+---------+-------+
| Host            | char(60)      |      | PRI |         |       |
| Db              | char(32)      |      | PRI |         |       |
| User            | char(16)      |      | PRI |         |       |
| Select_priv     | enum('N','Y') |      |     | N       |       |
| Insert_priv     | enum('N','Y') |      |     | N       |       |
| Update_priv     | enum('N','Y') |      |     | N       |       |
| Delete_priv     | enum('N','Y') |      |     | N       |       |
| Create_priv     | enum('N','Y') |      |     | N       |       |
| Drop_priv       | enum('N','Y') |      |     | N       |       |
| Grant_priv      | enum('N','Y') |      |     | N       |       |
| References_priv | enum('N','Y') |      |     | N       |       |
| Index_priv      | enum('N','Y') |      |     | N       |       |
| Alter_priv      | enum('N','Y') |      |     | N       |       |
+-----------------+---------------+------+-----+---------+-------+
13 rows in set (0.00 sec)
```

Une autre table, **host**, détermine quels sont les hôtes autorisés à se connecter à une base de données. La table **host** se présente comme suit :

```
mysql> describe host ;
+-----------------+---------------+------+-----+---------+-------+
| Field           | Type          | Null | Key | Default | Extra |
+-----------------+---------------+------+-----+---------+-------+
| Host            | char(60)      |      | PRI |         |       |
| Db              | char(32)      |      | PRI |         |       |
| Select_priv     | enum('N','Y') |      |     | N       |       |
| Insert_priv     | enum('N','Y') |      |     | N       |       |
| Update_priv     | enum('N','Y') |      |     | N       |       |
| Delete_priv     | enum('N','Y') |      |     | N       |       |
| Create_priv     | enum('N','Y') |      |     | N       |       |
```

© copyright Éditions Eyrolles

```
| Drop_priv        | enum('N','Y') |      |     | N       |       |
| Grant_priv       | enum('N','Y') |      |     | N       |       |
| References_priv | enum('N','Y') |      |     | N       |       |
| Index_priv       | enum('N','Y') |      |     | N       |       |
| Alter_priv       | enum('N','Y') |      |     | N       |       |
+-----------------+--------------+------+-----+---------+-------+
12 rows in set (0.01 sec)
```

Contrôle d'accès

Une fois que l'on a configuré les privilèges, les utilisateurs et les noms d'hôte auxquels on accorde l'accès, chaque connexion sera évaluée en vue de déterminer si elle sera autorisée. MySQL cherche d'abord le nom d'utilisateur et l'hôte. Quand une connexion s'opère, tous les paramètres sont comparés aux entrées de la liste. La comparaison est effectuée du début à la fin, jusqu'à ce qu'une correspondance soit trouvée. Si ce n'est pas le cas, la demande est rejetée. Pour que la vérification soit exhaustive, il faut s'assurer que la liste est triée correctement, car la base de données ne vérifie pas tous les cas et s'arrête à la première correspondance.

Si la connexion est autorisée, le logiciel vérifie que l'on a accès à la base de données — ou aux bases — à laquelle on essaie d'accéder. Il consulte les tables **host** et **db** pour vérifier si l'hôte a bien accès à la base de données souhaitée. À la différence de la vérification précédente, les tables seront consultées en fonction du type d'accès souhaité.

Pour les requêtes de type administration, il ne vérifie que la table des utilisateurs, car c'est la seule table où se situent les privilèges de l'administrateur. Si l'utilisateur a le privilège, l'action est autorisée, sinon elle est rejetée. Les requêtes qui sont liées à une base de données et qui manipulent les lignes d'une table sont d'abord vérifiées dans la table des utilisateurs : si l'utilisateur n'a pas le privilège adéquat, le logiciel vérifie alors dans les tables **host** et **db** si l'autorisation doit être accordée ou non ; si l'utilisateur ne figure pas dans la table **db**, la demande est rejetée ; si le champ **host** de l'utilisateur n'est pas renseigné, il détermine, par une référence croisée avec la table **host**, si l'accès doit être accordé.

Il suffit de gérer une simple liste de contrôle d'accès, pour spécifier les utilisateurs qui peuvent se connecter à une base de données, et depuis quels hôtes. C'est extrêmement efficace. On peut ainsi exclure toute intervention provenant d'un domaine externe, et n'autoriser que les utilisateurs d'un réseau local ou dotés d'adresses IP spécifiées.

© copyright Éditions Eyrolles

Installation du système

Pour que la base de données fonctionne en permanence, on doit initialiser le système de manière à ce qu'il démarre la base de données automatiquement lors du boot. On l'indique dans le répertoire **/etc/rc.d** de la machine. Si l'on a installé la distribution RPM, c'est déjà fait. On peut créer un fichier pour initialiser l'environnement ainsi que le nom des chemins et charger le démon de la base de données. Pour ce faire, nous allons créer un script **rc**. Créer des scripts **rc** exige le privilège root et ne peut être fait que par l'administrateur.

Créer d'abord un nouveau script, appelé **mysql,** dans le répertoire **rc**. Se placer dans **/etc/rc.d/init.d** et créer le fichier **mysql.** Ici, nous allons initialiser l'environnement de la base de données. Tout script démarre en demandant au système quel shell utiliser pour lancer **execute** :

```
# !/bin/sh
```

Ensuite, on initialise les noms de chemins et de répertoires :

```
PATH=/sbin:/usr/sbin:/bin:/usr/bin
basedir=/my/path/mysql
pid_file=/my/path/mysql/var/mysqld.pid
mysql_daemon_user=mysql
export PATH
```

Puis on écrit la partie de démarrage du script : il faut aller dans le répertoire **mysql** spécifié précédemment ; s'assurer que le programme **safe_mysqld** existe, puis l'exécuter ; spécifier l'utilisateur et le **pid_file** à utiliser, défini auparavant. Le fichier **pid** contient l'id du processus, et sera utilisé aussi pour arrêter le processus, dans la deuxième partie du script.

```
mode=$1

cd $basedir

case "$mode" in
  'start')
    # Start daemon
    if test -x bin/safe_mysqld
    then
      # Startup database
      bin/safe_mysqld --user=$mysql_daemon_user --pid-file=$pid_file &
    else
      echo "Can't execute $basedir/bin/safe_mysqld"
    fi
     ;;
```

© copyright Éditions Eyrolles

La dernière partie arrêtera la base de données en se servant du fichier **pid** généré auparavant.

```
'stop')
  # Stop daemon using the pid file
  if test -f "$pid_file"
  then
    mysqld_pid=`cat $pid_file`
    echo "Stopping database with pid $mysqld_pid"
    kill $mysqld_pid
  else
    echo "Cannot find $pid_file."
  fi
   ;;
```

Enfin, on donne quelques spécifications de syntaxe avant de sortir :

```
 *)
   # usage
   echo "usage: $0 start|stop"
   exit 1
   ;;
esac
```

Ensuite, ce script sera placé dans **/etc/rc.d/init.d/mysql**. Pour s'assurer qu'il est bien pris en compte lors de l'amorce, créer un lien symbolique, dans chacun des répertoires de niveau d'exécution, pour le démarrage et l'arrêt :

- dans les répertoires **rc0.d**, **rc1.d** et **rc6.d**, exécuter la commande :

```
shell$ ln –s. ./init.d/mysql K90mysql
```

- et dans les répertoires **rc2.d**, **rc3.d**, **rc4.d** et **rc5.d :**

```
shell$ ln –s. ./init.d/mysql S90mysql
```

Cela assurera le démarrage du serveur de base de données lors du boot et son arrêt pendant les phases de redémarrage ou d'arrêt du serveur. Comme cela peut se produire à différents niveaux d'exécution, il faut un lien dans chaque répertoire de niveau. L'administrateur se doit de connaître les niveaux d'exécution ; pour plus d'informations à ce sujet, voir la documentation de la distribution Linux. D'autres applications importantes mettant en œuvre la base de données peuvent aussi être démarrées de cette manière.

© copyright Éditions Eyrolles

Performance

Les performances de MySQL peuvent être optimisées lors de la compilation : le flag –**static** rend la base de données le plus rapide possible ; de même, se servir de sockets UNIX et non de connexions réseau.

La configuration matérielle peut aussi améliorer les performances : stocker la base de données sur des disques multiples pour diminuer le temps d'attente des mouvements de tête et de recherche des pistes, puisque plusieurs disques font le travail au lieu d'un seul. Cette technique est connue sous le nom de *striping*. Pour un striping à plus grande échelle, il faut se servir d'un RAID (*Redundant Array of Independent Disks*) qui donne de meilleurs résultats.

Les liens symboliques contribuent aussi aux gains, en établissant des liens vers des répertoires, dans des distributions placées sur d'autres lecteurs. On peut encore supprimer les fichiers d'échange, car cela évite que le système d'exploitation ne perturbe les accès à la base de données en bloquant les têtes de lecture pour les échanges entre la mémoire et les disques.

Le démon MySQL lui-même peut être réglé, grâce à de nombreuses variables. On les gère à l'aide de l'outil **mysqladmin.**

```
shell$. /mysqladmin variables
+---------------------------+---------------------------------------+
| Variable_name             | Value                                 |
+---------------------------+---------------------------------------+
| back_log                  | 5                                     |
| connect_timeout           | 5                                     |
| basedir                   | /home/bschwab/mysql/                  |
| datadir                   | /home/bschwab/mysql/var/              |
| delayed_insert_limit      | 100                                   |
| delayed_insert_timeout    | 300                                   |
| delayed_queue_size        | 1000                                  |
| join_buffer               | 131072                                |
| flush_time                | 0                                     |
| key_buffer                | 8388600                               |
| language                  | /home/bschwab/mysql/share/mysql/english/|
| log                       | OFF                                   |
| log_update                | OFF                                   |
| long_query_time           | 10                                    |
| low_priority_updates      | OFF                                   |
| max_allowed_packet        | 1048576                               |
| max_connections           | 100                                   |
| max_connect_errors        | 10                                    |
| max_delayed_insert_threads | 20                                   |
| max_join_size             | 4294967295                            |
| max_sort_length           | 1024                                  |
| max_write_lock_count      | 4294967295                            |
| net_buffer_length         | 16384                                 |
```

© copyright Éditions Eyrolles

```
| pid_file                  | /home/bschwab/mysql/var/obsidian.pid    |
| port                      | 3306                                    |
| protocol_version          | 10                                      |
| record_buffer             | 131072                                  |
| skip_locking              | ON                                      |
| skip_networking           | OFF                                     |
| socket                    | /tmp/mysql.sock                         |
| sort_buffer               | 2097144                                 |
| table_cache               | 64                                      |
| thread_stack              | 65536                                   |
| tmp_table_size            | 1048576                                 |
| tmpdir                    | /tmp/                                   |
| version                   | 3.22.30                                 |
| wait_timeout              | 28800                                   |
+---------------------------+-----------------------------------------+
```

Voici la définition de ces variables :

Tableau 7–2
Variables de contrôle des performances

Variable	Signification
backlog	nombre de requêtes qui attendent d'être traitées : c'est la taille de la file pour les connexions TCP/IP entrantes ; elle dépend beaucoup du système d'exploitation
concurrent_inserts	ON (par défaut) autorise les INSERT sur les tables MyISAM (sur disque) pendant les requêtes SELECT
connect_timeout	délai, en secondes, pour interrompre une connexion réseau inactive
delayed_insert_timeout	temps d'attente avant de supprimer une requête INSERT DELAYED
delayed_insert_limit	temps d'attente des SELECT après une requête INSERT DELAYED
delay_key_write	par défaut, **delay_key_write** est autorisé sur CREATE TABLE, ce qui accélère l'écriture des clés
delayed_queue_size	taille de la file d'attente en nombre de lignes pour INSERT DELAYED
flush_time	délai, en secondes, de mise à jour physique de toutes les tables
init_file	fichier d'initialisation chargé au démarrage

© copyright Éditions Eyrolles

Tableau 7–2
Variables de contrôle des performances

Variable	Signification
interactive_timeout	temps, en secondes, pendant lequel la base de données laissera une connexion inactive ouverte avant de la fermer
join_buffer_size	taille du buffer pour les jointures de table, alloué seulement aux jointures ; utilisé pour accélérer les jointures
key_buffer_size	taille du buffer pour les blocs d'index ; l'augmenter améliore la gestion des index
long_query_time	incrémente le compteur **slow_queries** quand la requête dure plus longtemps que la valeur indiquée
max_allowed_packet	taille maximale d'un paquet ; utilisé pour protéger la base de données contre les connexions erronées ; doit être augmentée pour les colonnes BLOB
max_connections	nombre maximum de connexions simultanées
max_connect_errors	nombre maximum d'erreurs au-delà duquel la connexion d'un hôte est désactivée
max_delayed_threads	nombre de threads pour INSERT DELAYED
max_join_size	taille maximale d'une jointure
max_sort_length	taille fixée au tri quand on se sert de valeurs BLOB ou TEXT
max_tmp_tables	pas encore implémenté
net_buffer_length	taille du buffer pour les demandes autres que les requêtes
net_retry_count	nombre d'essais avant l'abandon d'une connexion
record_buffer	taille du buffer pour un thread de parcours séquentiel de table
skip_show_databases	interdit aux utilisateurs de voir les tables de la base de données pour des raisons de sécurité
sort_buffer	taille du buffer du thread de tri

© copyright Éditions Eyrolles

Tableau 7–2
Variables de contrôle des performances

Variable	Signification
table_cache	nombre de tables ouvertes par les threads ; consomme des descripteurs de fichiers
tmp_table_size	taille des tables temporaires
thread_concurrency	nombre de threads simultanés
thread_stack	taille de la pile d'un thread
wait_timeout	même chose que **interactive_timeout**

Problèmes

Selon la distribution Linux, MySQL peut présenter quelques problèmes lors de l'exécution. MySQL fonctionne au mieux avec les versions les plus récentes de **glibc**, la bibliothèque C sur laquelle repose la base de données. MySQL demande au moins une version 2.0.7 de **glibc**, et la plupart des problèmes sont relatifs aux versions antérieures de cette bibliothèque. On trouve le numéro de la version installée sur le système à l'aide de la commande :

```
shell$ rpm –q glibc
```

Si l'on utilise une distribution binaire, on n'a pas à se préoccuper de la version de cette bibliothèque. Les distributions les plus anciennes se servent de **libc**, prédécesseur de **glibc**. Si la version de **libc** est supérieure ou égale à 5.4.46, cela fonctionne. Pour Red Hat 4.X, il y a un problème dans la résolution des noms d'hôtes avec des versions plus anciennes de **libc**. Il suffit de mettre à jour **libc** en version 5.4.46 ou supérieure. Pour Red Hat 5.X, il faut installer le **glibc** le plus récent, et les RPM suivants conviennent :

```
glibc-2.0.7-19
glibc-devel-2.0.7-19
```

La mise à jour des bibliothèques nécessite le privilège root : si vous ne l'avez pas, demandez à votre administrateur système d'installer, ou de mettre à jour, les dernières versions de ces bibliothèques.

© copyright Éditions Eyrolles

Threads

MySQL se sert des threads sous Linux. Les threads sont de petites parties de programme qui permettent de donner à la machine plus de travaux à traiter qu'un processus unique. Si le dernier **glibc** est installé, cela doit fonctionner sans avoir à installer de nouveaux programmes. Il peut être téléchargé à http://www.mysql.com/Downloads/Linux.

Cependant, avec les versions **glibc** 2.1.1 ou antérieures, un problème se produit avec les commandes **INSERT DELAYED.** Un correctif est disponible à http://www.mysql.com/Downloads/Patches/glibc-pthread_cond_timedwait.patch.

La version 3.23.7 de MySQL, en développement, contourne provisoirement ce problème.

Environnement d'exécution

Une fois MySQL installé, on dispose d'outils variés pour rassembler les informations concernant la base de données. Le premier, **mysqladmin**, exécute la plupart des tâches de l'administrateur dont on a besoin : création et suppression de bases de données, écriture physique (flush) des tables, changement des mots de passe utilisateurs, rechargement, ainsi que l'arrêt ou la recherche du statut de la base de données.

Aller d'abord dans le répertoire **bin** de l'installation MySQL :

```
shell$ cd /my/path/mysql/bin
```

Si l'on exécute **mysqladmin** sans aucun argument, on obtient une liste complète de toutes ses caractéristiques :

```
shell$. /mysqladmin
```

Pour trouver la version :

```
shell$. /mysqladmin version
./mysqladmin Ver 7.15 Distrib 3.22.30, for pc-linux-gnu on i686
TCX Datakonsult AB, by Monty

Server version          3.22.30
Protocol version        10
Connection              Localhost via UNIX socket
UNIX socket             /tmp/mysql.sock
Uptime:                 10 min 33 sec

Threads : 1 Questions : 3 Slow queries : 0 Opens : 6 Flush tables : 1 Open tables : 2
```

© copyright Éditions Eyrolles

Pour trouver l'état :

```
shell$. /mysqladmin status
Uptime : 901 Threads : 1 Questions : 4 Slow queries : 0 Opens : 6 Flush tables : 1
Open tables : 2
```

On peut consulter toutes les variables d'environnement avec :

```
shell$. /mysqladmin variables
```

Pour créer une première base de données, utiliser **mysqladmin**. L'option –**p** demande le mot de passe :

```
shell$. /mysqladmin –p create MYDATABASE
```

Une fois créée, elle peut être supprimée avec la commande :

```
shell$. /mysqladmin –p drop MYDATABASE
```

Pour arrêter la base de données correctement, utiliser enfin **mysqladmin** :

```
shell$. /mysqladmin –p shutdown
```

On peut importer et exporter facilement des données de MySQL en se servant de **mysqldump** et **mysqlimport**. Ces outils permettent de faire un *dump* complet de la base de données, qui pourra être réimporté plus tard dans la base de données. C'est donc une manière pratique de sauvegarder toutes les données de la base.

D'autres outils sont proposés dans la distribution MySQL. Le tableau 7-3 ci-dessous résume toutes les fonctionnalités de **mysqladmin**.

Tableau 7–3
Outils disponibles dans mysqladmin

Outils	Utilisation
myisamchk	vérifie, optimise, répare et décrit les tables MySQL
make_binary_release	crée une version binaire de MySQL à partir d'une version compilée
msql2mysql	programmes de conversion pour passer de **msql** à MySQL
mysqlaccess	vérification des privilèges d'accès aux enregistrements de la base de données pour les hôtes et les utilisateurs
mysqladmin	administration générale de MySQL
mysqlbug	mailing-list des rapports de bogues à MySQL

© copyright Éditions Eyrolles

Tableau 7–3
Outils disponibles dans mysqladmin

Outils	Utilisation
mysqld	démon réel de la base de données MySQL
mysqldump	exporter une base de données dans un fichier
mysqlimport	importer un fichier dans la base de données
mysqlshow	affiche des informations à propos des bases de données et des tables
mysql_install_db	script utilisé au début de l'installation pour initialiser les tables initiales
replace	script pour remplacer des chaînes ; utilisé par **msql2mysql**
safe_mysqld	script qui maintient le démon **mysqld** en fonctionnement, et le relance quand il meurt

Dépannage

En cas d'incident sur le serveur de base de données MySQL, il existe un fichier de trace dans le répertoire des données avec une extension .**err**. De manière générale, il contient l'historique des erreurs. S'il n'y a aucune information concernant la cause de la panne, on peut lancer **mysqld** avec l'option d'exécution –**log** qui ajoute des informations relatives à l'activité de la base de données, ou l'option –**debug**, qui donne encore plus d'informations.

Quelques points à vérifier :

- Avoir la version la plus récente de la distribution.
- Avoir **glibc**, ou **libc**, le plus récent (selon la distribution).
- Vérifier les permissions des utilisateurs sur tous les fichiers appartenant à l'utilisateur de base de données.
- Vérifier le statut avec **mysqladmin** pour s'assurer que, lors du chargement, il ne s'est rien passé d'anormal.
- Problèmes de connexion : s'assurer que le contrôle d'accès est initialisé correctement pour l'utilisateur, le mot de passe et le nom d'hôte, et que le serveur fonctionne réellement (on peut le vérifier avec la commande **ps).**
- S'assurer qu'aucun autre programme n'utilise les mêmes ports que MySQL.

© copyright Éditions Eyrolles

- Si un thread est mort, détruire tous les autres threads avant de redémarrer la base de données.
- Essayer la connexion localement et à distance. Si la connexion à distance échoue, vérifier que les options réseau sont actives. Si la connexion locale échoue, s'assurer que les sockets UNIX sont actifs.
- Vérifier qu'il y a assez de connexions pour tous les clients.
- Vérifier qu'il y a suffisamment de mémoire pour la base de données : les besoins en ressources des très grandes bases augmentent rapidement.
- Vérifier que les options de performance sont réglées correctement pour la taille des tables et que les tables ne sont pas trop grandes.
- Vérifier qu'aucun changement significatif ne s'est produit dans le fonctionnement de la base de données depuis l'impression de cet ouvrage. Se reporter à la documentation fournie avec MySQL.

Résumé

De prime abord, l'installation de MySQL intimide, mais en revoyant les options et en vous formant sur ce qui convient le mieux à votre besoin, vous pouvez mettre d'emblée en service la base de données. MySQL est proposé dans des distributions variées : binaire, RPM et source. Les distributions binaires sont adaptées à un type particulier de plate-forme et ne peuvent être utilisées sur aucune autre. Elles sont relativement faciles à installer mais peuvent exiger un accès root pour être installées dans les répertoires au niveau du système. Les distributions RPM sont les plus faciles à manier, mais seulement si l'on dispose de Red Hat Linux. Ces distributions effectuent même une partie de la configuration à votre place. Les dernières distributions, recommandées, sont les distributions source. Elles sont proposées avec une documentation et des exemples abondants, et peuvent être personnalisées et adaptées à vos besoins.

MySQL est constamment en développement, comme on peut le voir d'après le nombre de niveaux des distributions. Les distributions stables sont recommandées non seulement pour leur stabilité, mais encore pour la gamme de caractéristiques testées qu'elles proposent. Les distributions toutes neuves sont destinées à tester les nouvelles caractéristiques qui vont enrichir la base de données. Les anciennes versions désapprouvées ne sont pas utiles, sauf si vous en êtes utilisateur. Maintenir à jour une distribution stable, aujourd'hui la version 3.22, offrira toutes les caractéristiques et les garanties quant à la stabilité exigée d'une application.

© copyright Éditions Eyrolles

L'installation réelle dépend de la distribution choisie. Les installations RPM et binaires exigent un minimum de travail et peuvent être réalisées assez vite. Les distributions RPM nécessitent Red Hat et le privilège root. Les distributions binaires exigent d'être installées dans un répertoire système. Il n'y a aucune compilation à faire. La distribution source n'exige pas d'accès root, sauf si on la place dans un répertoire système. Cette distribution exige plus d'outils système que les distributions précédentes, mais ils sont déjà installés par la distribution Linux. La seule étape supplémentaire est la compilation, dont la durée dépend des ressources système.

Une fois installée, il faut configurer et initialiser la base de données. La première étape de l'installation est de s'assurer qu'une sécurité correcte est mise en place. Rien n'est plus important, et il faut notamment s'assurer que seuls les utilisateurs habilités y ont accès, en particulier si l'information qui y est stockée est sensible. Même si ce n'est pas le cas, ne laissez pas aux utilisateurs la possibilité d'abîmer les données ou de les supprimer.

Le premier utilisateur créé, une fois la base de données installée, est l'utilisateur root. Le mot de passe de cet utilisateur doit être modifié immédiatement après l'installation : sinon, n'importe qui peut se connecter et disposer d'un accès complet à la base de données. Créer aussi un utilisateur sur le système qui exécute le démon MySQL. Ne l'exécutez pas sous le compte root, car ce serait la porte ouverte aux pirates qui essaient d'exploiter le démon pour obtenir le privilège root. Une fois que cet utilisateur est créé, les fichiers et permissions doivent être gérés de telle sorte qu'il soit le seul à accéder à ces fichiers, et à pouvoir les écrire.

Avec MySQL il y a beaucoup de types de privilèges, qui déterminent ce qu'un utilisateur peut faire dans la base de données : les privilèges de l'administrateur, tels que la fermeture ou le rechargement de la base de données ; ceux relatifs aux bases de données et aux tables, tels que la possibilité de lire les tables et d'y insérer de nouvelles données. Les nouveaux utilisateurs seront créés avec des rôles bien spécifiques. Un utilisateur peut être créé avec seulement les fonctions d'administrateur. Les utilisateurs normaux qui manipulent les données auront des permissions seulement pour ces opérations et n'auront pas accès aux opérations les plus importantes. Les droits des utilisateurs peuvent aussi être restreints en fonction de l'origine de leur connexion à la base de données. On peut créer des utilisateurs qui sont autorisés à se connecter localement, ou à partir d'hôtes distants précis, ou les deux. Il existe des tables spéciales qui gèrent l'accès restreint à certaines tables et d'autres, certains hôtes, que l'on peut combiner pour établir une matrice de privilèges très spécifiques de l'environnement. Ces privilèges sont attribués par la commande **GRANT** ou, si l'on connaît bien SQL, par des commandes SQL qui s'appliquent directement aux trois tables où sont gérés les droits.

© copyright Éditions Eyrolles

La base de données mise en service doit être aussi disponible que possible. La méthode de démarrage par défaut du démon MySQL est **safe_mysqld**, un script qui suit l'état du démon de la base de données : si la base de données a été interrompue ou arrêtée, il la redémarre automatiquement. Il y a une autre méthode pour s'assurer que le démon MySQL est démarré lors du boot : en créant, dans le répertoire **/etc,** un script **rc** que l'on peut lier aux différents niveaux d'exécution de la machine, on s'assure que lorsque la machine est redémarrée, MySQL est démarré avec les autres programmes du serveur.

Les dernières étapes de l'installation concernent l'optimisation de la distribution. On peut optimiser la version compilée si l'on a une distribution source. L'avantage de disposer des sources est que l'on peut compiler avec différentes options pour tirer le meilleur parti de la plate-forme. Il existe aussi des méthodes, au niveau du matériel, pour accélérer la base de données. Enfin, il y a des flags d'exécution spécifique du démon de la base de données MySQL qui permettent de régler et de modifier les variables pour accélérer encore la base de données. Une fois que vous avez terminé l'installation, créé les utilisateurs de base de données, et même essayé d'améliorer la performance de la base de données, vous vous sentirez dans votre élément.

FAQ

Q. Comment exécuter plusieurs serveurs MySQL sur ma machine ?

R. Cela peut être fait de deux manières.

Avec la distribution source, vous disposez de flags de configuration spécifiques qui autorisent la base de données à pointer vers différents emplacements et à utiliser des ports différents :

```
shell$. /configure  –with-tcp-port=port_number \
        –with-unix-socket=file_name \
        –prefix=/usr/local/mysql-3.22.9
```

Vous pouvez aussi utiliser, en l'absence de distribution source, les flags d'exécution de **mysqld** :

```
shell$ /my/path/bin/safe_mysqld –socket=file_name –port=port_number
```

Il ne faut pas avoir deux démons différents capables de modifier la même base de données. Le chemin peut être spécifié par –**datadir**.

© copyright Éditions Eyrolles

Q. Comment puis-je répliquer ma base de données ?

R. Pour l'instant, MySQL n'a pas de fonction pour le faire automatiquement. Un moyen simple est de faire chaque nuit un dump de la base de données et de l'importer dans une autre base. La fréquence de cette opération dépend bien sûr de la fraîcheur souhaitée pour les données dupliquées.

Q. Comment puis-je sauvegarder ma base de données ?

R. La commande **mysqldump** permet de sauvegarder toutes les tables de la base de données.

```
shell$ mysqldump –tab=/save/path –opt –full
```

On peut aussi arrêter le serveur et archiver le répertoire de la base de données avec **tar**. Pour recharger les tables de la base de données, on rétablit les fichiers dans le répertoire de base de données avec **untar**, ou bien on restaure le dump.

Q. Comment puis-je migrer vers une version plus récente si j'ai déjà installé MySQL ?

R. Vous pouvez arrêter la base de données courante et la copier dans un répertoire de nom différent, puis installer la nouvelle distribution comme la première fois. Ensuite, transférez le répertoire de la base de données dans le nouveau répertoire et redémarrez votre base de données.

Q. Que se passe-t-il si j'oublie mon mot de passe ?

R. Il faut d'abord arrêter le serveur, puis le relancer avec l'option **–skip-grant-tables** pour ne *pas* utiliser la table des privilèges, se connecter à nouveau à la base de données puis, soit utiliser **GRANT** pour créer le nouveau mot de passe, soit utiliser des commandes SQL pour renseigner directement le mot de passe dans la table des utilisateurs.

Q. Comment puis-je gérer des transactions ?

R. À l'inverse des grandes bases de données, MySQL ne sait pas gérer les transactions. Il faut attendre les prochaines versions.

© copyright Éditions Eyrolles

8

Installer et gérer Progress sur Linux

Dans ce chapitre

- Installation
- Configuration
- Tester la base de données
- Dépannage
- Forces et faiblesses de Progress

© copyright Éditions Eyrolles

Introduction

La base de données Progress est intégrée, littéralement, dans des milliers de paquetages de taille moyenne des plus répandus aujourd'hui (par exemple, MFG/Pro de qad Inc., Symix, Unifi de FiServ et Trend de NxTrend). En fait, Progress vient en tête du marché des bases de données intégrées (voir http://www.progress.com/services/pressrm/releases/pr_gartner_041499.htm pour les détails) et c'est l'un des plus grands acteurs de l'émergence, sur le marché, de l'ASP (*Association of Shareware Professionals*). Les paquetages qui reposent sur le système de gestion de base de données relationnelle Progress sont utilisés par 60 % des entreprises, figurent au classement de Fortune 500 et rapportent plus de 2,5 milliards de dollars de revenu annuel aux associés de Progress. Et pourtant Progress Software Corporation (PSC) n'a pas la notoriété à laquelle on pourrait s'attendre pour une société dont le chiffre d'affaires annuel se situe autour de 300 millions de dollars et qui a réalisé plus de 80 000 installations dans une centaine de pays.

Ce phénomène est peut-être dû à la stratégie de « marketing furtif » retenue par PSC, qui consiste à vendre principalement par un réseau de revendeurs. Progress a toujours fondé ses programmes de ventes sur la vente aux développeurs de logiciels plutôt que sur les ventes directes aux grands distributeurs d'informatique. Cela s'est avéré extrêmement fructueux depuis 15 ans puisqu'il existe maintenant une plage de plus de 4000 applications développées à l'aide de la base de données et du L4G. Mais, comme PSC doit éviter d'entrer en conflit avec ses propres développeurs-revendeurs, le nom de beaucoup de partenaires de Progress est plus connu que Progress lui-même.

Qu'importe la stratégie de vente, Progress, en tant que plate-forme, a beaucoup à offrir. La base de données a toutes les caractéristiques d'une solution OLTP (système transactionnel) à grande échelle, et présente déjà l'avantage de ne pas avoir besoin d'un administrateur de base de données à temps plein pour fonctionner. Progress s'adapte aisément aux systèmes, depuis les PC individuels avec 32 Mo de RAM aux plus grands serveurs multiprocesseurs de IBM, HP, Sun, Compaq et DEC. Le L4G est un langage complet qui n'exige aucun code L3G pour couvrir tout le champ d'une application. Plus de 90 % des applications ne contiennent que du code L4G et, grâce à l'environnement d'application runtime, sont totalement portables sur toute grande plate-forme UNIX, dont Linux.

Au premier trimestre 2000, PSC propose la version 8.3 de Progress pour Linux. C'est la plate-forme de déploiement la plus avancée, actuellement utilisée par la plupart des paquetages à base de Progress. La version la plus récente de Progress, la version 9, sera disponible sur Linux quand la version 9.1 sera publiée, vers la fin du premier trimestre 2000. Les versions antérieures à la version 8.3 ne sont pas natives et fonctionnent sur Linux à l'aide du

© copyright Éditions Eyrolles

paquetage iBCS et du portage de Progress pour SCO OpenServer. Les versions 6.2 à 8.2 ont été utilisés sur Linux de cette manière. Toutefois, ces versions n'entrent pas dans le cadre de cet ouvrage et on trouvera, sur le sujet, des ressources Internet telles que les FAQ Progress sur Linux à l'adresse http://marathon-man.com/pl/Progress-Linux-FAQ.html, et le *Progress Email Group* (PEG—www.peg.com). Ce chapitre est consacré à la version 8.3, son installation, sa configuration et sa gestion.

Installation

L'installation de Progress est un processus simple en trois étapes : installer d'abord les exécutables à partir du média, changer tous les paramètres du noyau, puis installer l'environnement. Après l'installation, on est prêt à créer une première base de données.

Installer à partir du média

L'installation à partir du média consiste à copier tout simplement des exécutables depuis la bande ou le CD-ROM fournis par Progress. Selon la plate-forme, les scripts d'installation fonctionnent de manière légèrement différente. Pour Linux, comme pour la plupart des plates-formes UNIX, on dispose d'une version sur bande où chaque section de la bande contient un produit. Le script d'installation de la bande est un enchaînement de commandes **cpio,** pour extraire les fichiers, et **dd,** pour sauter les produits non utilisés. La version sur CD-ROM est un peu différente puisque les fichiers sont disponibles sans extraction. Dans ce cas, Progress se sert de son propre schéma d'archivage de fichier pour les combiner dans un seul fichier d'archive sur le CD-ROM.

Avant de commencer l'installation réelle, il faut s'assurer que l'on est prêt.

Conditions préalables

L'installation du média Progress est un processus simple qui exécute les scripts fournis sur le CD-ROM. Avant de commencer, il faut s'assurer que l'on a approximativement 100 Mo d'espace libre pour le logiciel. Selon les produits achetés, on peut avoir besoin de moins d'espace disque. S'assurer également de la licence, « Progress Software Corporation License Addendum », fournie avec les produits. On l'appelle familièrement la *feuille verte*, d'après la couleur du papier sur laquelle elle est imprimée.

© copyright Éditions Eyrolles

Nous postulons, pour la suite de l'exposé, que vous disposez d'un ordinateur fonctionnant déjà sous Linux, équipé d'un lecteur CD-ROM opérationnel. Si l'on souhaite accéder à la base de données en mode client-serveur, on a besoin de TCP/IP ; les terminaux passifs n'en ont pas besoin. Progress a été officiellement testé sur Red Hat Linux version 6.1, mais fonctionnera sur la plupart des distributions offrant un noyau stable 2.2.12 ou supérieur. Le processus décrit ici a été testé sur la version 6.1 de Mandrake, similaire à Red Hat.

Installer

Placer le CD-ROM dans le lecteur et le monter. La commande exacte varie selon la distribution mais, en général, elle est de la forme :

```
mount /dev/cdrom /mnt/cdrom
```

Vérifier que le répertoire courant se trouve sur autre chose que le CD-ROM. Si on oublie cela, le script d'installation rappelle qu'il ne peut s'exécuter directement depuis le CD-ROM. Exécuter l'utilitaire d'installation **proinstall.** Un écran d'accueil s'affiche (figure 8.1) : appuyer simplement sur la touche **Entrée p**our passer aux données de configuration de produit (figure 8.2).

```
/mnt/cdrom/proinst
                    +----------------------------------+
                    |              Welcome             |
                    +----------------------------------+
                    |                                  |
                    | WELCOME TO THE PROGRESS SOFTWARE |
                    |       INSTALLATION UTILITY       |
                    |                                  |
                    |          Release 8.3B20          |
                    |     Copyright (c) 1992-1999      |
                    |     Progress Software Corp.      |
                    |      All Rights Reserved.        |
                    |                                  |
                    |                                  |
                    |            [Enter=OK]            |
                    +----------------------------------+
```

Figure 8–1
Écran d'accueil de l'installation

C'est le cœur de l'installation — où l'on entre les numéros de série et les clés du produit. Les touches de fonction sont décrites sur le côté droit ; utiliser la touche **Tab** pour naviguer entre les champs. Entrer d'abord le nom de la société, ou un autre identifiant unique. Cela n'est pas essentiel — il n'est utilisé nulle part ailleurs que dans l'installation. Se servir de la feuille verte pour entrer le numéro de série du premier produit. Le numéro de version sera déjà rempli, appuyer simplement sur **Tab** pour passer au champ suivant. On

© copyright Éditions Eyrolles

```
+--------------------------------------------------------------------+
|                    Product Configuration Data                      |
+--------------------------------------------------------------------+
|                                                [Enter=Additional]  |
| Company Name: Geoff Crawford_________________  [Ctrl-E=Done]       |
| Serial Number: 0019281                         [CTRL-T=Quit]       |
| Release Number: 8.3B_____________              [CTRL-N=Release Notes]|
| Control Number: GTM?9 77UWX X9?MR              [CTRL-V=View]       |
|                                                [TAB=Next Field]    |
|                                                                    |
|                                                                    |
+--------------------------------------------------------------------+
```

Figure 8–2
Saisie des numéros de produit et des codes de contrôle

doit ensuite entrer les trois ensembles de codes de contrôle du numéro de série. Bien vérifier les codes et appuyer sur **Entrée**. Si les codes de contrôle concordent avec le numéro de série, l'écran indique que l'entrée est valide, sinon il faut rechercher les erreurs.

Répéter la saisie des numéros de série pour chaque produit indiqué dans la feuille verte. Il n'y a aucune raison de ne pas installer tous les produits en même temps. Quand le dernier numéro de série est saisi, appuyer sur **Ctrl-E** pour continuer l'installation détaillée. Certains émulateurs de terminal (par exemple, TinyTERM de Century Software) utilisent **Ctrl-E** pour des fonctions internes — il faut modifier cela si l'on se sert d'un tel émulateur. Attention aussi à l'option **Quit** (**Ctrl-T**) qui signifie l'annulation de l'installation entière.

```
+------------------------------+
| Type, Device and Destination |
+------------------------------+
| Select Type of Installation  |
| Select Destination Pathname  |
| Continue with Installation   |
| View Release Notes           |
| Cancel                       |
| Quit Installation            |
+------------------------------+
```

Figure 8–3
Options d'installation

On est alors invité à confirmer que l'on installe tous les produits : répondre simplement **Y.** L'écran suivant, intitulé **Type, Device, and Destination**, est représenté à la figure 8.3. Dans les installations à partir d'un CD-ROM, il n'y a besoin d'aucune information supplémentaire pour le périphérique (*device*), il suffit d'entrer le type et la destination. Utiliser les flèches pour sélectionner **Type of installation** et appuyer sur **Entrée**. Utiliser encore les flèches pour

© copyright Éditions Eyrolles

choisir **Custom installation** : l'installation personnalisée n'est pas difficile, elle demande seulement quelques valeurs supplémentaires ; elle est recommandée dans tous les cas.

Choisir ensuite **Destination Pathname** et indiquer la valeur par défaut **/usr/dlc**. Rien n'est mauvais en soi, concernant le chemin ou le nom du répertoire, mais il est souvent suggéré d'avoir un nom unique pour chaque version de Progress. Choisir un nom tel que **/opt/progress/dlc83b** pour la version de Progress, car il est courant d'installer, plus tard, une version mise à jour. Ne jamais installer une nouvelle version dans le même répertoire qu'une ancienne — choisir un autre répertoire et régler les variables d'environnement en conformité. Bien que, la plupart du temps, cela ne concerne que les grandes installations et ne soit pas obligatoire, on obtiendra des gains de performance en plaçant les exécutables sur leur propre lecteur de disque. Dans tous les cas, modifier le nom de répertoire de manière à se trouver dans l'ensemble des zones juxtaposées nécessaires à la préparation.

Ce peut être le moment de consulter les notices relatives aux versions. Par ailleurs, on les trouve dans un fichier texte sur le CD-ROM. Sélectionner l'option **Continue with Installation**. On est alors invité à choisir une langue (figure 8.4) : utiliser les flèches pour sélectionner la langue principale. Quand on appuie sur la touche **Entrée,** sur la sélection en surbrillance, un astérisque marque la langue sélectionnée. Choisir alors l'option **Make Default** pour la première langue. Terminer en sélectionnant les autres langues souhaitées, puis choisir **Continue with Installation**.

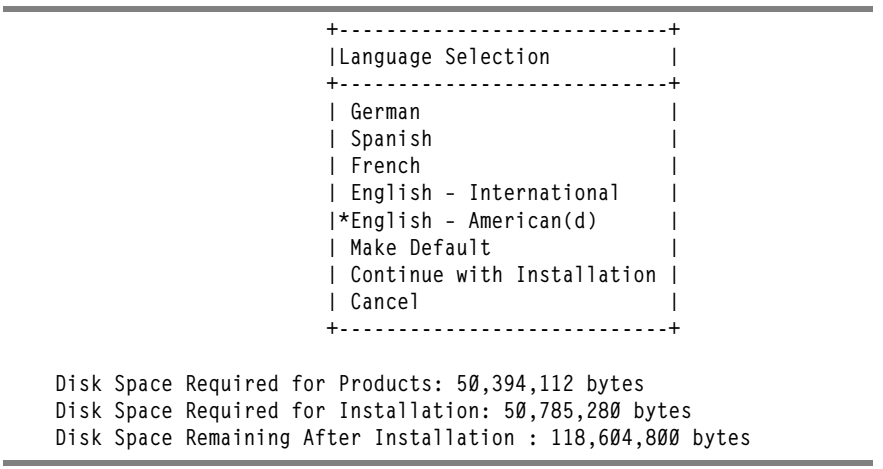

Figure 8–4
Sélection de la langue

Comme il s'agit d'une installation personnalisée, il vous est maintenant proposé d'installer seulement des parties de chaque produit, ou encore

© copyright Éditions Eyrolles

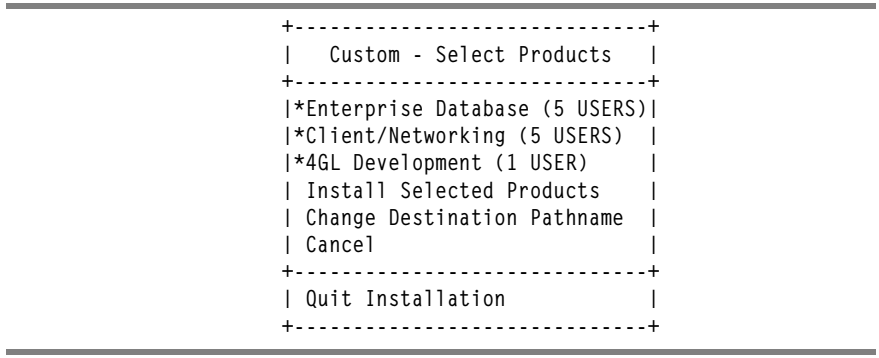

Figure 8–5
Sélection des produits

d'installer certains utilitaires optionnels (figure 8.5). Tous les produits entrés dans le premier écran apparaissent ici, avec un astérisque qui indique qu'ils seront installés. En sélectionnant un produit et en appuyant sur la touche **Entrée**, on obtient une liste complète des paquetages compris dans le produit. En particulier, il est conseillé de choisir tous les paquetages, sauf si l'on manque d'espace disque. La plupart des paquetages sont déjà sélectionnés par défaut, mais il y en a un, en particulier, qui peut se révéler utile. Ceux qui migrent leurs systèmes vers la version native Linux de Progress doivent savoir que les utilitaires de migration des bases de données version 8.2 ne sont pas activés par défaut, mais qu'ils doivent être installés. Ils font partie des options proposées avec les produits de déploiement de base de données tels que *Enterprise Database*, *WorkGroup Database*, ou *Stand Alone Database*.

Après avoir vérifié tous les produits et confirmé les sélections, choisir **Install Selected Products** à l'aide des flèches et appuyer sur **Entrée**. L'utilitaire d'installation demande de confirmer que l'on est prêt à commencer l'installation ; entrer **Y** pour oui. Une dernière question se rapporte à la copie des scripts dans **/usr/bin** (figure 8.6). C'est une survivance de l'époque où **/usr/bin** était l'endroit approprié où ranger de tels scripts.

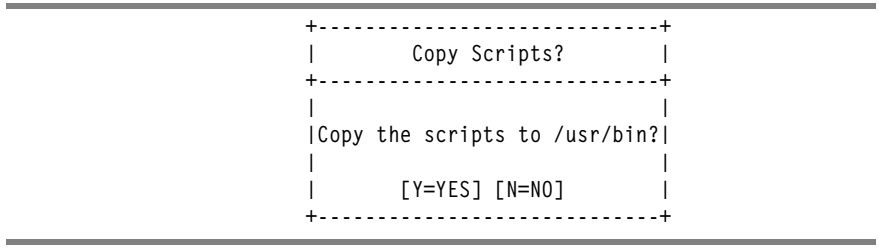

Figure 8–6
Copie des scripts de shell, en option

© copyright Éditions Eyrolles

Il vaut mieux répondre **Non** et ajouter **$DLC/bin** au PATH des utilisateurs qui ont accès à ces scripts (c'est-à-dire les programmeurs et les administrateurs — les utilisateurs finaux n'ont pas besoin d'accéder à la base de données à l'aide de ces scripts).

Ensuite, Progress commence à décompresser les fichiers du CD-ROM. Selon la vitesse du système et le nombre de produits, cela prend habituellement entre 5 et 15 minutes. On reçoit enfin la confirmation que l'installation du média est terminée.

Paramètres du noyau

Maintenant que tous les produits Progress ont été copiés depuis le CD-ROM, il se peut que le noyau Linux ait besoin d'être configuré. Progress se sert de deux ensembles principaux de ressources contrôlées par le noyau Linux — la mémoire partagée et les sémaphores. Avant de décider des valeurs exactes de chaque paramètre, il est bon de faire un petit retour sur ces ressources, pour mieux les comprendre et les initialiser correctement.

La mémoire partagée sert à attribuer de la RAM destinée aux accès multi-utilisateur. Au lieu d'écrire toutes les informations de la base de données directement sur le disque, où chaque utilisateur provoquera une opération d'entrée/sortie lente, Progress se sert de la RAM, plus rapide, comme d'une sorte de dépôt d'enregistrements de la base de données. Les blocs sont lus depuis le disque dans la zone de mémoire partagée (connue aussi en tant que *buffer pool*, ou cache) où ils sont alors disponibles non seulement pour l'utilisateur qui a demandé les données, mais pour tous les utilisateurs du système. Le moteur de base de données Progress est multithread et tire profit des SMP, sous réserve qu'il y ait des processeurs multiples. Comme plusieurs utilisateurs peuvent tenter simultanément de modifier la mémoire partagée, Progress se sert, si besoin est, d'une autre ressource, les sémaphores, pour verrouiller les parties touchées de la mémoire partagée, et pour assurer que ces opérations sur la mémoire partagée sont atomiques. En réalité, les sémaphores sont seulement utilisés pour les opérations « lentes » ; les verrouillages mutex[1], ou les *latches*, sont utilisés pour les opérations « rapides ».

Les blocs de données modifiés sont éventuellement sauvegardés sur le disque et éliminés de la mémoire — mais pas toujours immédiatement. Si cela vous semble dangereux, rassurez-vous : le système de base de données a été conçu solide comme un roc. Bien que Oracle ait la réputation d'être à l'épreuve des pannes intempestives, Progress a en réalité le meilleur score dans cette caté-

1. mutex (*Mutual Exclusion Service*) : qui opèrent l'exclusion réciproque ; *latches* : verrous de type bascule. *N.d.T.*

© copyright Éditions Eyrolles

gorie, si l'on se fie à des études menées par des utilisateurs, tel le Rapport annuel du VAR[2] Business.

Progress a incorporé des techniques qui gèrent la concurrence, la restauration après une panne, ainsi que des modèles de gestion de données qui garantissent l'intégrité de la base de données en conservant les blocs fréquemment utilisés dans le pool, en vue d'une réutilisation (ce qui élimine partiellement de coûteuses opérations d'entrée/sortie). Les détails de ces modèles sont complexes et sortent du cadre de cet ouvrage. Pour plus d'informations, consulter la section des documents techniques du PEG (*Progress Email Group*), à http://www.peg.com. Les architectes du moteur de base de données, désignés couramment comme l'*équipe moteur*, ont écrit plusieurs excellents livres blancs, qui expliquent le fonctionnement interne de la base de données.

Voici maintenant une liste de paramètres, et de valeurs à leur affecter. La plupart d'entre elles reposent soit sur le nombre d'utilisateurs du système, soit sur le nombre de bases de données. Les fichiers d'en-têtes **sem.h** et **shm.h** contiennent la description de ces paramètres dans le noyau Linux. Ces deux fichiers se trouvent généralement dans **/usr/src/linux/include/linux**.

Mémoire partagée

Les trois paramètres à initialiser sont :

- **SHMMNI** : nombre d'identifiants de mémoire partagée ;
- **SHMSEG** : nombre de segments de mémoire partagée par processus ;
- **SHMMAX** : taille maximale d'un segment de mémoire partagée.

La mémoire partagée est organisée en segments ; chaque groupe de segments reçoit un identificateur unique. Chaque base de données Progress a besoin d'au moins un segment de mémoire partagée et d'un identifiant. Prendre le nombre de bases de données que l'on envisage d'exécuter sur le système, et l'arrondir, par sécurité, au multiple de cinq le plus proche — le résultat est une valeur minimale pour **SHMMNI**. La valeur par défaut dans Red Hat Linux est de 10, ce qui sera suffisant pour la plupart des systèmes.

SHMSEG et **SHMMAX** sont liés entre eux : comme la mémoire partagée est organisée en segments, la taille maximale de la mémoire partagée est le produit du nombre de segments par la taille de chaque segment. Les versions 8.3 de Progress créent des segments dont la taille est le minimum de la valeur de **SHMSEG**, soit 16 776 214 octets (16 Mo) : il est donc inutile, actuellement, de donner à **SHMSEG** une valeur plus grande que 16 Mo, car Progress ne créera pas de segments de taille supérieure.

2. VAR : *value-added resellers*, revendeurs à valeur ajoutée. *N.d.T.*

© copyright Éditions Eyrolles

Autre question : de combien de mémoire partagée doit-on disposer ? Les manuels donnent des valeurs précises mais, invariablement, c'est le cache qui prédomine sur tout autre usage, pour l'utilisation de la RAM et l'évaluation de la mémoire partagée. Pour des raisons de performance, dans un système de production, il faut essayer d'affecter au cache un minimum absolu d'environ 2 Mo d'espace par utilisateur. Si possible, essayer d'avoir 4 Mo, quitte à acquérir plus de mémoire, sachant que la mémoire partagée doit être comprise dans la RAM physique. Si l'on alloue plus de mémoire partagée que l'on n'a de RAM physique, Linux placera une partie des pages du cache dans l'espace d'échange pour combler la différence, et cela aura un impact désastreux sur les performances.

Pour atteindre la capacité optimale de **SHMSEG** : prendre le nombre d'utilisateurs de chaque base de données et le multiplier par la valeur de la RAM que l'on a décidé de lui réserver ; répéter ce calcul pour chaque base de données et déterminer laquelle aura la valeur maximale de mémoire partagée ; diviser la valeur maximale de la RAM par 16 Mo, valeur indiquée pour **SHMMAX**, et l'arrondir à la valeur supérieure. Le résultat donne le nombre maximum de segments exigé par une base de données : ce sera la valeur de **SHMSEG**.

Sémaphores

Trois autres paramètres sont à initialiser :

- **SEMMNI** : nombre d'identifiants de sémaphore ;
- **SEMMNS** : nombre de sémaphores pour le système ;
- **SEMMSL** : nombre de sémaphores par identificateur.

Les sémaphores, comme la mémoire partagée, sont organisés en un ensemble de noms doté d'un identifiant. Progress a besoin d'un ensemble de sémaphores par base de données. Le calcul de **SEMMNI** est le même que celui de **SHMMNI** — les deux paramètres ont généralement la même valeur.

Progress se sert d'environ cinq sémaphores pour son propre usage interne, plus un sémaphore pour chaque utilisateur connecté à un instant donné à la base de données. **SEMMSL** doit contenir le nombre maximum d'utilisateurs connectés à la base de données, à supposer qu'ils le soient tous simultanément : lui donner une valeur égale au nombre maximum d'utilisateurs dans la base de données plus cinq, et ajouter 10 % par sécurité.

SEMMNS, dans la plupart des noyaux Linux, contient en général le produit **SEMMNI * SEMMSL**. Ce n'est pas mauvais en soi, mais cela suppose que chaque utilisateur se connecte à chaque base de données. S'il n'y a qu'une base de données, on n'a pas besoin d'un autre calcul, mais, en cas de bases de données multiples, cela peut conduire à un gaspillage. Dans le doute, laisser le calcul tel quel. Mais si l'on envisage de créer une très grande base de

© copyright Éditions Eyrolles

données ainsi qu'une très petite, telle une base de données de développement avec seulement cinq utilisateurs, à comparer aux 400 utilisateurs du système de production, il vaut mieux modifier **SEMMNS**. Une bonne règle empirique est d'utiliser **promon** dans chaque base de données pour trouver le nombre maximum d'utilisateurs par base et d'additionner tous ces nombres. Ajouter, au total, cinq utilisateurs par base, puis 10 % par sécurité.

Dans certaines variantes d'UNIX, il existe aussi des structures de sémaphores Undo et une map des sémaphores (**SEMUME/SEMMAP**) mais, dans Linux, elles ne sont plus utilisées.

Autres paramètres divers

En plus de la mémoire partagée et des sémaphores, il y a d'autres paramètres à prendre en considération :

- **MAXUMEM** : maximum de mémoire utilisée par un processus ;
- **NOFILES** : nombre maximum de fichiers ouverts par processus ;
- **NFILES** : nombre maximum de fichiers ouverts sur le système ;
- **NBUFS** : nombre de tampons disque (*buffers*).

Puisque tous les utilisateurs, ainsi que le processus de base de données, doivent se connecter au segment de mémoire partagée, il arrive que la mémoire à laquelle chaque processus Linux doit accéder augmente considérablement. **MAXUMEM** doit être égal à la taille de l'image de l'exécutable Progress, augmentée de la taille de la zone de mémoire partagée la plus grande dans le système. Une bonne règle est de prendre **SHMSEG * SHMMAX** (ce qui donne la plus grande valeur possible de mémoire partagée) et d'ajouter 5 Mo, ou de lui donner la taille de la RAM physique du système.

Comme nous le verrons plus loin, certaines bases de données sont en réalité composées de seulement deux ou trois fichiers du système d'exploitation. Cependant, les plus grandes bases de données, qui se servent de structures multivolumes, sont réparties dans de nombreux fichiers du système d'exploitation. Quand on utilise de telles bases de données, il faut parfois modifier les paramètres **NOFILES**, **NFILES** et **NINODES**. Avec la version 8.3 de Progress, le nombre de fichiers d'une base de données est de 256, au maximum. (La version 9 peut se servir de milliers de fichiers, aussi on prendra garde d'améliorer ce calcul quand la version 9 sera disponible sur Linux.) Chaque utilisateur a besoin des trois fichiers standard entrée, sortie, erreur, ainsi que de tout fichier spool ou temporaire qu'il crée (à concurrence de 12 pointeurs de fichier, sans compter les fichiers relatifs à la base de données). Par conséquent, **NOFILES** n'a pas besoin d'être supérieur à 300.

NFILES est le nombre de fichiers sur le système : il sera, au maximum, égal au nombre d'utilisateurs connectés, multiplié par le nombre de bases de données

© copyright Éditions Eyrolles

et multiplié par le nombre de fichiers du système d'exploitation de chaque base de données. Attention : il ne s'agit ici que du nombre de fichiers relatifs à Progress. On doit y ajouter le nombre de fichiers ouverts par Linux lui-même et par les autres utilisateurs de Linux. Les manuels Progress suggèrent de se servir de ce nombre maximal pour **NFILES** mais, comme l'indique le calcul, ce nombre peut rapidement devenir très grand. On peut s'en tenir là, mais cela suppose que chaque utilisateur est connecté à toutes les bases de données. En réalité, des nombres plus petits pourront convenir.

Sur de petits sites où il n'y a qu'une seule base de données, ce n'est pas un problème en général, mais sur de plus grands sites, cela conduira à un gaspillage. Un calcul correct pour **NFILES** fera la somme des produits du nombre d'utilisateurs simultanés par le nombre de fichiers du système d'exploitation de chaque base de données, pour définir le niveau d'utilisation de Progress. On pourra ajouter 10 % pour faire face à la croissance ou à des modifications de profil des logins. Ajouter encore 10 % pour l'usage des fichiers du système d'exploitation.

Il reste à voir un dernier paramètre Linux. Depuis que Progress se sert de son propre cache pour gérer les accès aux enregistrements, les tampons disque standard de Linux n'ont plus la même utilité. Sur les systèmes où la RAM est mesurée, il est souvent avantageux de réduire légèrement les tampons Linux et d'allouer la RAM ainsi dégagée à la mémoire partagée. Le paramètre **NBUFS** contrôle le nombre de tampons, mais il faut prendre des précautions avec ce paramètre. Tous les fichiers de spool, les fichiers de programme et les autres fichiers du système d'exploitation doivent passer par les tampons disque de Linux. Diminuer cette valeur peut donner de meilleures performances du point de vue de la base de données, mais causer un ralentissement de l'impression et d'autres accès disque. Dans le doute, réduire ce paramètre graduellement depuis la valeur d'origine. Vérifier soigneusement si le système présente des ralentissements à l'impression ou des pauses quand de nouveaux programmes Progress sont lancés. Se servir des outils d'optimisation de Progress pour équilibrer les paramètres et voir comment une augmentation du cache affecte les performances de la base de données. On trouvera plus de détails sur l'équilibrage de l'allocation de la RAM à la section de configuration, dans la suite de ce chapitre.

Installation de l'environnement

Pour initialiser l'environnement Linux pour Progress, il suffit de fixer quelques variables d'environnement et, éventuellement, de modifier les hôtes TCP/IP et les fichiers des services — tâches qui doivent être familières à tout administrateur de système Linux. Si l'on utilise une application en mode caractère, il faudra aussi modifier la définition du terminal dont se sert

© copyright Éditions Eyrolles

Progress. Cette installation de terminal s'ajoute à l'installation normale **termcap/terminfo**, et peut se révéler plus complexe que le simple choix du bon type de terminal.

Variables d'environnement

Progress se sert de deux variables d'environnement Linux : **DLC** et **TERM**. La variable **DLC** doit pointer vers le répertoire où a été installé Progress. Normalement, elles auront les valeurs suivantes (en **Bash**) :

```
DLC=/opt/progress/dlc83b
TERM=linux
Export DLC TERM
```

On doit opérer avec le profil de login système (**/etc/profile**) ou bien dans le **.profile** ou le **.login** de chaque utilisateur (ou ce qui convient au shell utilisé). Sauf s'il y a une raison impérative de faire une installation personnalisée, toutes les autres variables d'environnement de Progress doivent avoir des valeurs par défaut compatibles avec **$DLC**. Il vaut toujours mieux mettre les exécutables Progress dans le **PATH**. Ajouter les deux répertoires **$DLC** et **$DLC/bin** au **PATH** de tous les administrateurs de base de données, de manière à ce que tous les utilitaires de la base de données soient dans leur chemin.

Installation du terminal — PROTERMCAP

Progress se sert d'une bibliothèque d'avertissements et de messages (*curses-like*)[3] appelée *Vermont Views* (que l'on reconnaît à ses messages d'erreur préfixés par **vv**_) pour l'affichage et le rafraîchissement des écrans : la valeur **TERM** doit donc être initialisée correctement. Progress a son propre fichier de configuration de type **termcap** appelé **PROTERMCAP**, qui définit certains attributs du terminal, non disponibles dans le **termcap** standard. Ils concernent principalement les touches de fonction étendues et les options de couleur du L4G Progress.

Les définitions relatives à de nombreux terminaux sont fournies dans le fichier **PROTERMCAP,** livré avec le produit. En ce qui concerne Linux, il y a une entrée pour la console Linux sous le type de terminal **linux** (celle où les codes de contrôles réduisent la zone de données visible). Pour un terminal X, le type qui convient est **xterm**. Les anciens terminaux de type « toute visibilité » ou *space-taking* (Wyse 50, Televideo 925, et ainsi de suite) ne sont plus supportés. On peut toujours chercher une hypothétique émulation de terminal. La disposition d'origine du clavier d'un VT100 n'avait que quatre touches de fonctions, et un VT220 un usage prédéfini dans le matériel pour

3. *curse* : juron ; dans ce cas, réaction du système sous forme d'un message écrit à l'écran. *N.d.T.*

© copyright Éditions Eyrolles

les cinq premières touches de fonction. Bien que certains émulateurs accordent des touches supplémentaires, Progress n'a que les touches « réelles » dans le fichier **PROTERMCAP.** On peut soit modifier le fichier **PROTERMCAP**, soit utiliser une émulation qui a un ensemble complet de touches de fonction.

Les entrées spécifiques de **PROTERMCAP** sont décrites en détail dans les manuels techniques de Progress et sortent du cadre de cet ouvrage. Mais voici quelques conseils généraux d'utilisation de **PROTERMCAP**. Laisser le fichier de configuration d'origine tel quel et, au lieu de le modifier, en faire une copie et initialiser une variable d'environnement Linux, appelée **PROTERMCAP**, qui contient le nom du chemin complètement qualifié du nouveau fichier **PROTERMCAP (**c'est un cas où il est bon de personnaliser et d'écraser les options par défaut).

Deux autres problèmes peuvent se poser à la configuration du terminal : la touche espace arrière (*backspace*) et l'interruption Linux. Progress s'attend à ce que le caractère espace arrière soit **Ctrl-H**, même sur les terminaux DEC où il est défini en réalité par **Ctrl-?** (c'est-à-dire, la touche **Delete).** Progress s'attend aussi à ce que la touche de suppression soit une fonction de suppression de caractère, ce qui interfère avec certaines options où la touche de suppression du terminal est programmée en tant que touche d'interruption. Dans les deux cas, utiliser la commande **stty** dans le **.profile** de l'utilisateur pour changer ces valeurs. **Ctrl-C** est généralement admis comme touche alternative d'interruption et il est compatible avec les définitions de terminal fournies dans Progress :

```
stty erase ^h intr ^c
```

Installation du mode client-serveur

Si la seule méthode d'accès est l'émulation d'un terminal telnet, il s'agit de ce que l'on appelle un *client self-service*, et il n'y a pas d'autres étapes à franchir. Le mode client-serveur diffère de l'utilisation de telnet pour accéder à la machine Linux. Si les utilisateurs accèdent à la base de données en mode client-serveur, il faut un autre fichier. Progress a besoin de l'adresse TCP/IP de la machine et d'un port TCP/IP dédié pour chaque base de données atteinte *via* le réseau. L'adresse IP de la machine est en principe affectée pendant l'installation de TCP/IP. Pour réserver le port TCP/IP, éditer le fichier **/etc/services**. Chaque ligne a le format :

```
name 1000/tcp
```

Créer une ligne par base de données, en remplaçant **nom** par un identifiant unique propre à chaque base de données. On peut utiliser le nom réel de la base de données comme identifiant, ce qui le rend plus facile à identifier.

© copyright Éditions Eyrolles

Choisir des numéros de port supérieurs au dernier numéro en cours d'utilisation et laisser au moins un écart de 500 entre les ports. Ce n'est pas le seul port utilisé en réalité et, dans le modèle d'assignation de port de Progress, des incompatibilités peuvent quelquefois se produire, si les numéros de port affectés aux bases de données multiples sont trop proches. Un seul port TCP est obligatoire ; la partie base de données de Progress n'utilisera pas d'UDP.

Configuration

Configurer une base de données Progress n'est pas difficile. Il y a quelques décisions à prendre et la base de données réelle pourra être créée. D'abord, examinons les structures de répertoire dont se sert Progress, puis nous débattrons de l'arbitrage entre espace et vitesse du disque.

Structure de répertoire

La configuration de la structure des répertoires pour une base de données Progress est une partie importante de la préparation, en amont du déploiement d'une application. Bien entendu, on peut se contenter de tout mettre en vrac dans un seul grand répertoire, sans plus s'en soucier. Cette solution n'est pas seulement inélégante, mais elle présente l'inconvénient de grever l'administration des données. Une structure de répertoire bien conçue et organisée est essentielle pour qu'un site fonctionne bien. En général, on crée des systèmes de fichiers séparés, correspondant à certains usages de la base de données et de l'application. De cette manière, on peut disposer les systèmes de fichiers sur différents disques configurés et en régler finement les options, telles que le type **fs,** la taille de bloc, le type RAID et, quand on se sert des fichiers de journaux, beaucoup d'autres aspects du système de fichiers. Pour ce faire, on conseille la structure suivante (tableau 8.1) :

Tableau 8–1
Tableau des répertoires

Nom de répertoire	Usage
/opt/progress/dlc83b	Établir des répertoires distincts par version de Progress : les mises à jour, les tests et les rollbacks éventuels seront beaucoup plus faciles.
/mnt/db	Les fichiers **.db, .st** et **.d#** sont tous ici. Ce sera un groupe de disques agrégés par bande[a] et mis en miroir, d'une taille de bloc **fs** adaptée à la taille de bloc **db** et dotés des options maximales de fiabilité et de récupérabilité.

© copyright Éditions Eyrolles

Tableau 8–1
Tableau des répertoires

Nom de répertoire	Usage
/mnt/bi /mnt/ai	Les systèmes de fichier **.bi** et **.ai** seront sur des disques en miroir, non agrégés par bande, dotés de caractéristiques de système de fichiers visant la fiabilité et la récupérabilité maximales.
/mnt/logs	Établir un système de fichier pour les fichiers **lg** et les autres données de surveillance. Recopier les logs quotidiens dans ce système de fichiers toutes les nuits. Il n'a pas besoin d'être en miroir ou agrégé par bande — des disques anciens conviennent. Mettre en œuvre une stratégie d'archivage qui place tous les fichiers dans des répertoires ayant pour nom la date **aaaa.mm.jj** de manière à avoir un historique utilisable. Archiver les fichiers **.ai** de la même manière.
/mnt/source /mnt/rcode /mnt/tmp /mnt/scripts	Ces systèmes de fichiers seront rapides et fiables mais ils n'ont pas besoin d'être récupérables au plus haut degré — l'agrégat par bande (*striping*) et la mise en miroir suffisent. Le *full logging* (une option de journalisation du **fs,** actuellement non disponible sur Linux) n'est pas nécessaire puisque l'on peut tout récupérer à partir de la sauvegarde, mais la fiabilité est importante, car les sessions d'utilisateur peuvent tomber en panne en cas d'indisponibilité des fichiers rangés sur ces systèmes de fichiers.
/mnt/recover	Délimiter une zone pour récupérer les sauvegardes — dans la plupart des scénarios de restauration, il est indiqué de conserver la base de données endommagée pour le cas où la sauvegarde serait endommagée, elle aussi, ou bien mal faite. Ce système de fichiers aura les mêmes caractéristiques que le système de fichier **db.**
/mnt/dump	Zone réservée pour décharger et charger, ou sauvegarder sur disque (avant une sauvegarde sur bande, bien sûr), séparée de toutes les autres zones, ce qui procure un maximum de souplesse. Le mieux est que cette zone se serve de disques qui ne sont pas utilisés en dehors d'un processus de déchargement, de chargement ou de sauvegarde.

a. striped : agrégat par bandes (strip) ; les blocs sont écrits en utilisant les disques de l'ensemble successivement, ce qui donne, avec un peu d'imagination, un aspect de bandes. *N.d.T.*

Même si l'on n'a qu'une base de données petite ou peu utilisée, qui ne semble pas justifier l'usage de disques multiples et le réglage individuel des systèmes de fichiers selon la méthode de création et d'appellation de systèmes de fichiers qui précède, on peut toujours mettre en place cette organisation comme une structure de répertoire simple et monter les systèmes de fichiers selon les besoins : cela facilite les futures adaptations et évolutions. Cette conception est une garantie de succès des migrations futures.

© copyright Éditions Eyrolles

Espace disque et débit d'entrée/sortie

Un autre élément à prendre en considération quand on configure un système en vue d'exécuter une base de données : la disposition des disques physiques et les capacités du sous-système disques. Cette section ne concerne pas les petits systèmes peu chargés.Si l'on a suivi le modèle de conception proposé dans ce qui précède, on pourra revenir en arrière pour appliquer ces techniques si le besoin s'en fait sentir.

La capacité d'un disque, en giga-octets, ou le flux de transfert, en méga-octets par seconde, est un indicateur intéressant si l'on fait de la vidéo à la demande ou d'autres applications à dominante texte pré-Internet. Mais ce dont il faut vraiment se préoccuper est le nombre d'*opérations* traitées par seconde. Bien sûr, le PC a convaincu tout un chacun que la seule chose vraiment importante est la *taille* d'un lecteur de disque, mais si un jour on doit expliquer pourquoi on « gaspille » autant d'espace à un comptable particulièrement pointilleux, il va falloir être très patient.

Le sous-programme d'entrée/sortie de la base de données Progress gère les données en blocs uniformes dispersés, au hasard sur le disque. Le disque qui contient la base de données doit, par conséquent, se concentrer sur des entrées/sorties aléatoires. Les agrégats par bandes (*striped*) sont excellents pour l'optimisation des E/S dans ce cas. La mise en miroir a des résultats corrects. RAID 5 est un mauvais choix s'il n'y a pas d'activité d'écriture sensible. Pourtant une nouvelle forme de RAID, RAID 10 (qui combine RAID 0 + RAID 1) est le choix idéal. JBOD (*Just a Bunch Of Disks*) vient en deuxième position, le troisième est celui d'un disque assez grand, et RAID 5 sera évité à tout prix.

La gestion des entrées/sorties des fichiers d'images « avant » (**.bi**) et « après » (**.ai**) est surtout séquentielle. Les disques exécutent les opérations séquentielles 10 ou 20 fois plus vite que les opérations aléatoires. Si l'on mélange les types d'accès, on obtient des performances d'accès aléatoires. Si la demande globale pour les opérations d'entrées/sorties est plus grande que les disques ne peuvent en traiter, cela crée une file d'attente sur le disque et cause de mauvaises performances. En premier lieu, ce problème se fait sentir au niveau des checkpoints (trop fréquents, comme dans le cas de synchronisations répétées entre les copies de données en mémoire et le disque) qui gèle tous les autres utilisateurs si l'on a une licence *workgroup* (ou si l'on ne se sert pas des caractéristiques d'écriture asynchrone de pages de la licence *enterprise).* Dédier un disque au fichier **.bi** permet d'éviter d'être affecté par d'autres activités d'entrée/sortie et de fonctionner à pleine vitesse. C'est particulièrement important avec la licence *workgroup*, mais encore plus important avec la licence *enterprise* quand on commence à pousser les capacités de la machine.

© copyright Éditions Eyrolles

Dédier un disque au fichier **.ai** est important pour les même raisons, et pour une autre encore : les images « après » constituent un moyen de reprise avec restauration actualisée. Si les fichiers **.ai** sont sur le média qui a été endommagé lors de l'événement à l'origine de la restauration, on est en présence d'un cercle vicieux. Autant que possible, consacrer un disque aux images « après ». Si l'on ne peut pas, on n'aura qu'une protection limitée contre les pannes matérielles sérieuses et les erreurs de l'administrateur (par exemple, le reformatage d'un disque). On se sert encore des images « après » — qui protègent aussi d'erreurs, beaucoup plus probables, des utilisateurs, par exemple, supprimer tous les clients — pour effectuer des reprises à intervalles réguliers. Mais on ne tire tous les avantages de cette fonctionnalité que si le fichier se trouve sur un disque distinct.

La mise en miroir des fichiers **.ai** et **.bi** est une bonne idée — pour ne perdre ni l'un ni l'autre, sous aucun prétexte. Mais cela ne signifie pas le remplacement, ni de l'un ni de l'autre. Le fichier **.bi** est essentiel à la restauration après une panne totale. Comme la mise en miroir reproduit fidèlement les erreurs humaines telles que le scénario précédemment évoqué de la « suppression de tous les clients » ou même un simple **rm *** malencontreux dans le mauvais répertoire, seules les images « après » peuvent vous aider à sortir de ces impasses. La mise en miroir améliore les temps mais ne protège de rien.

L'agrégat par bandes des fichiers **.bi** et **.ai** est inutile — il va à l'encontre de la nature séquentielle des entrées/sorties et peut même limiter le débit éventuel de ces fichiers. RAID 5 est, bien sûr, totalement inadapté à ces fichiers.

Les systèmes dotés d'une mémoire fortement contrainte ou chargés de requêtes mal indexées peuvent se trouver confrontés à des taux d'entrées/sorties très lourds, dans la zone de fichiers provisoires définie par le paramètre de démarrage de session –**T.** C'est un système de fichiers où Progress crée les fichiers temporaires destinés à divers usages. En principe, ces fichiers ne sont pas liés (*unlinked*) et sont par conséquent invisibles, mais ils peuvent être rendus visibles à l'aide du paramètre –**t** (minuscule). Les principaux types de fichiers trouvés dans la zone –**T** sont :

- **Srta<pid>** : le fichier de tri — stockage de la définition de la requête client et du r-code ;
- **Lbia<pid>** : l'image « avant » locale — stockage des sous-transactions ou des variables qui permettra les « undo » ;
- **DBIa<pid>** : zone de débordement des tampons de tables provisoires.

Si vous ne pouvez utiliser la mémoire pour rediriger ces entrées/sorties (les paramètres –**mmax** et –**Bt** sont les paramètres pertinents), il est important que la zone –**T** ait un excellent débit d'entré/sortie — ce que l'on trouve dans un ensemble de disques agrégés par bande et en miroir.

© copyright Éditions Eyrolles

Créer une nouvelle base de données

Il y a, certes, beaucoup de manières d'initialiser une base de données, mais nous allons nous intéresser à celles qui fonctionnent le mieux. Progress propose deux méthodes différentes pour mettre en oeuvre une base de données : volume simple et multivolume. La méthode multivolume est absolument nécessaire dans un environnement à grande échelle ; elle offre de surcroît beaucoup d'avantages à tous les utilisateurs.

Multivolume

Progress propose deux types d'organisation de bases de données. Les bases de données volume simple sont les plus faciles à créer mais ont une taille maximale de 2 giga-octets (la limite d'un appel système **lseek()**). Les fichiers de données ont également la même limite de 2 Go dans les bases de données multivolumes, mais ils peuvent être combinés pour disposer d'une plus grande capacité de stockage. Dès la version 9 de Progress, toutes les bases de données seront multivolumes, ce qui, associé aux améliorations prévisibles de performance face à la nécessité de gérer des fichiers plus grands, fait du multivolume le choix recommandé.

Les bases de données Progress sont réparties en quatre types de fichiers ou *extents* : image-avant, image-après, données et le fichier DB. Le fichier image-avant donne à Progress la possibilité d'annuler les transactions inachevées. Les fichiers image-avant sont identifiés par des extensions de fichier **.b1**, **.b2**, etc. ou seulement **.bi** pour une base de données simple volume. Les fichiers image-après se terminent par **.a1**, **.a2**, etc. et servent à la restauration actualisée. Les fichiers de données contiennent à la fois les données et les index. Chacun de ces fichiers est décrit dans un fichier de structure et prend sa place dans une collection qui forme la base de données. Le nom du fichier de structure finit par **.st** ; il sert à créer son équivalent binaire, le fichier **.db** — une fois que la base de données a été créée, le fichier **.st** n'est plus nécessaire mais peut être recréé à l'aide de l'utilitaire **prostrct** — et il est judicieux de le garder à titre de documentation.

Les étapes de création de nouvelles bases de données sont bien décrites dans la documentation Progress ; néanmoins, nous préférons proposer un exemple. Il faut d'abord créer un fichier **.st** contenant la liste de tous les fichiers, définis par leur type (image-avant, image-après, données), leur longueur, fixe ou variable, et s'ils sont fixes, leur taille en blocs de 1 Ko (ou ce que l'on crée comme taille de bloc de base de données — 1 Ko étant la valeur par défaut). Il ne peut y avoir qu'une image avant et un fichier de données de longueur variable. Pour rester simple, définissons un fichier de structure avec un fichier fixe pour l'image-avant et un fichier variable pour les données. Il est conseillé de gérer l'image-après car elle donne la capacité de régénérer la base de

© copyright Éditions Eyrolles

données dans son dernier état correct, même après la dernière sauvegarde. Elle n'est pas traitée dans l'exemple qui suit pour plus de simplicité. Le fichier de structure peut s'appeler, par exemple, **mydata.st** et ressembler à :

```
B /mnt/bi/mydata.b1 f 10000
B /mnt/bi/mydata.b2
D /mnt/db/mydata.d1 f 500000
D /mnt/db/mydata.d2
```

La première colonne indique le type de fichier : B pour image-avant, A pour image-après, D pour données ; le **f** signifie un fichier de taille fixe. Dans ce cas, si le fichier image-avant (B) a atteint la taille de 10 Mo (c'est-à-dire, 10 000 blocs), le second fichier **.b2** sera utilisé. De la même façon, le fichier **.d2** sera utilisé dès que les fichiers de données auront atteint la taille de 500 Mo. Remarquer aussi que les fichiers sont répartis entre différents points de montage.

Le calibrage des fichiers est un exercice d'équilibre. En général, plus il y a de fichiers, moins il se produit d'accès conflictuels ; il vaut donc mieux créer 10 fichiers de 200 Mo plutôt que deux de 1 Go, pour une base de données de 2 Go. On peut aussi réserver de l'espace de croissance pour réduire au minimum les interventions de l'administrateur — pour ajouter ou redimensionner les fichiers.

Il reste deux étapes pour terminer le processus. D'abord, l'utilitaire Progress **prostrct** sert à traduire le fichier de structure en une disposition physique de base de données prête à l'emploi. Tous les fichiers de longueur fixe sont créés avec les tailles requises (arrondies aux 16 Ko les plus proches). Voici la syntaxe, qui utilise le fichier de structure de base de données créé auparavant :

```
prostrct create /mnt/db/mydata mydata.st
```

Vérifier les fichiers des divers systèmes d'exploitation créés, suite à cette étape, **.db**, **.b1**, **.b2**, **.d1**, **.d2**, etc. Remarquer que le fichier **.db** n'est pas décrit dans le fichier de structure, mais qu'il en dérive. Le fichier **.db** est créé dans le répertoire spécifié par le troisième paramètre passé à la commande **prostrct : /mnt/db/mydata**. Tous ces fichiers constituent la base de données, mais sans l'étape finale, il n'existe encore aucune structure interne à la base de données : on dit qu'il s'agit d'une base de données *void*. Pour lui en donner une, on copie une base de données vide (*blank*) dans la base *void* à l'aide de la commande **procopy :**

```
procopy empty /mnt/db/mydata
```

La base de données vide est un cas particulier de la commande **procopy.** Il n'est pas nécessaire de spécifier le chemin complet de cette structure même si

© copyright Éditions Eyrolles

elle réside dans le répertoire DLC, où les exécutables Progress ont été installés. Progress trouvera la base de données vide à l'aide de la variable d'environnement **$DLC.** Il existe aussi plusieurs versions secondaires de la base de données vide, selon la taille des blocs. Pour créer une base de données avec une taille de bloc de 8 Ko, utiliser **empty8** au lieu de **empty** dans la commande **procopy.** Il faut se souvenir que la taille de bloc variable est une caractéristique de la licence *enterprise*. Toutes les bases de données *workgroup* ont une taille de bloc fixée à 1 Ko.

La base de données est maintenant disponible pour un accès en mode mono-utilisateur. Il faut démarrer un agent multi-utilisateur pour donner les autorisations nécessaires en mode multi-utilisateur. Utiliser la commande **proserve** pour démarrer cet agent. Sans autres options, l'agent autorisera 10 utilisateurs sur le système local à accéder à la base de données. Pour autoriser un plus grand nombre d'utilisateurs, utiliser l'option –**n.** Pour un accès en réseau, utiliser **–H –N –S**. Ces options, ainsi que nombre d'autres paramètres liés aux performances, seront décrits plus loin ; ils le sont aussi dans le manuel de référence de l'administration système de Progress.

Calibrage du cache

Le plus grand consommateur de RAM, dans la plupart des installations Progress, est le cache déterminé par le paramètre –**B.** C'est une grande zone de mémoire partagée utilisée comme cache des blocs de disque, en vue d'améliorer les performances.

Surtout, ne pas le régler en fonction du nombre d'utilisateurs, mais d'après le nombre d'entrées/sorties que le sous-système disque peut gérer. Au départ, essayer de situer les opérations d'entrée/sortie physiques en dessous d'un seuil tel que les disques soient utilisés entre 25 et 50 %, qu'il n'y ait aucune file d'attente et que le temps de service soit inférieur à 20 ms. Si les disques ont un temps d'accès moyen de 10 ms, ils peuvent exécuter 100 opérations d'entrée/sortie aléatoires par seconde et peuvent soutenir le rythme de 25 à 50 opérations d'entrée/sortie par seconde, sans risque de file d'attente.

Il est probable que le nombre d'utilisateurs aura une influence sur le nombre de demandes d'accès à la base de données, mais leur relation avec le taux d'échec est non linéaire. Elle suit une courbe de la forme :

R = L / sqrt((C * B)

où :

R = nombre d'opérations E/S physiques

L = nombre d'opérations E/S logiques

© copyright Éditions Eyrolles

Lectures logiques	-B	Nombre d'opérations d'entrée/sortie	Taux de succès	Taux d'échec
1 000 000	5 000	11 547	98.845%	1.155%
1 000 000	10 000	8 165	99.184%	0.816%
1 000 000	15 000	6 667	99.333%	0.667%
1 000 000	20 000	5 774	99.423%	0.577%
1 000 000	25 000	5 164	99.484%	0.516%
1 000 000	30 000	4 714	99.529%	0.471%
1 000 000	35 000	4 364	99.564%	0.436%
1 000 000	40 000	4 082	99.592%	0.408%
1 000 000	45 000	3 849	99.615%	0.385%
1 000 000	50 000	3 651	99.635%	0.365%
1 000 000	55 000	3 482	99.652%	0.348%
1 000 000	65 000	3 203	99.680%	0.320%
1 000 000	70 000	3 086	99.691%	0.309%
1 000 000	75 000	2 981	99.702%	0.298%
1 000 000	80 000	2 887	99.711%	0.289%

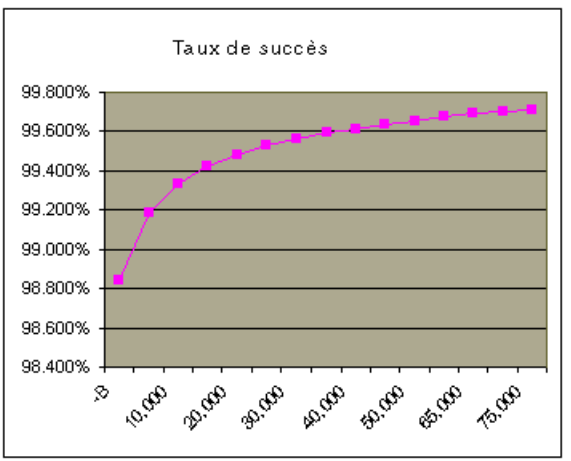

Figure 8–7
Exemple d'optimisation de la taille de la RAM, en fonction des entrées/sorties

© copyright Éditions Eyrolles

B = taille du cache

C = une certaine constante (autour de 1,5)

Ainsi, il n'y a pas vraiment de règle, pour déterminer la valeur X de RAM par utilisateur, que l'on peut appliquer ici. Mais pour assurer un démarrage en trombe, une valeur de 2 à 4 Mo par utilisateur est généralement une bonne estimation. Dès que l'on dispose d'un certain nombre de données, on peut régler plus précisément cette valeur pour une meilleure concordance avec les entrées/sorties. La figure 8.7 illustre un exemple.

Exécuter Progress

Le processus d'exécution quotidienne d'une base de données Progress ne représente pas, de manière générale, un travail à plein temps. En principe, on écrit, une fois pour toutes, des scripts de démarrage, d'arrêt et de sauvegarde de la base de données. Par la suite, on consacre surtout son temps à recueillir des données pour vérifier les performances du système, et éventuellement les optimiser. Beaucoup de systèmes fondés sur Progress fonctionnent pendant des années sans attention particulière d'un administrateur.

Gérer la base de données

Progress dispose de scripts simples pour démarrer, arrêter, surveiller et sauvegarder la base de données. Bien qu'ils soient généralement adaptés à un environnement de développement initial, une approche plus sophistiquée ou personnalisée est souvent souhaitable dans un environnement de production.

Une première étape est de créer un petit *script d'environnement*, qui concentre toutes les variables d'environnement utilisées couramment par Progress ou liées à l'application en un seul script : celui-ci est alors intégré, « à discrétion », dans les scripts qui en ont besoin. Par exemple :

```
# Common Environment Script
#
# To use :
#
# . /mnt/scripts/env

export TMP=/mnt/tmp
export MTH=`date '+%m''
export DAY=`date '+%d''
export LOGS=/mnt/logs/$MTH.$DAY
```

© copyright Éditions Eyrolles

```
export SRV=`uname `
export SCRIPTS=/mnt/scripts
export DLC=/opt/progress/dlc83b
export DBNAME=/mnt/db/sports

# standard messaging routine

msg()
{
  MSG=$1 ; shift
  echo "$MSG"
  echo `date` "$MSG" >> $LOGS/start.log

 # If additional arguments were passed then mail the message to
 # those people (the names might be pager addresses)

  [ "$1" != "" ] && {
  echo $SRV: `date` "$MSG" | mailx -s "$MSG" $*
  }
}
```

Démarrer la base de données

Les sites relativement petits utilisent souvent des scripts qui détectent automatiquement l'état de la base de données et la démarrent lorsqu'un premier utilisateur tente d'y accéder. Quelquefois, ils l'arrêtent même automatiquement quand le dernier utilisateur se déconnecte. Bien que très commode, cette approche recèle plusieurs pièges. Un tel script ne devra jamais supprimer le fichier **.lk.** Si Progress indique qu'il existe un fichier de verrouillage, il faut prendre ce message au sérieux et comprendre d'où il vient. Ce peut être parce qu'un autre utilisateur a, indépendamment de vous, démarré une session mono-utilisateur légitime (peut-être pour exécuter une opération de maintenance de base de données). Supprimer le fichier **.lk** puis accéder à la base de données sont des actions qui génèrent deux ensembles de processus, qui risquent de mettre à jour la base de données simultanément, sans avoir conscience l'un de l'autre — une situation qui conduit évidemment au désastre.

Autre inconvénient de cette approche, il est difficile d'obtenir un accès exclusif à la base de données pour les opérations de maintenance ou de restauration. Un conseil : dissocier le démarrage du serveur de base de données du démarrage des sessions utilisateur et instituer un ensemble de contrôles standard, respectés par tous les scripts, de manière à pouvoir administrer la base de données, en toute sécurité et à tout moment. Pour cela, il existe une technique bien utile, utiliser un fichier *flag* pour donner l'accès aux sessions utilisateur, comme ceci :

© copyright Éditions Eyrolles

```
# Start server
#

. /mnt/scripts/env

# Start something and verify it

strt()
{
  CMD=$1
  USR=$2
  GTX=$3
  WHO=$4

  msg "Starting: $1"
  $CMD >> $LOGS/start.log
  sleep 5
  ps –fu $USR | grep "$GTX" > /dev/null
  if [ $? = "1" ]; then
    msg "*** $CMD did not start" $WHO
    exit 0
  else
    msg "+++ $CMD started successfully"
  fi
}

# Create logging directory

if [ ! -d $LOGS ]; then
  mkdir $LOGS
  chmod 755 $LOGS
fi

msg “Starting database”
msg “Locking out users”

echo “Starting Database – Please wait” > $TMP/User_Lockout
chmod 644 $TMP/User_Lockout
rm -f $TMP/DB_IS_UP
sleep 5

strt “$DLC/bin/_mprosrv -pf $SCRIPTS/db.pf”
  progress
  “[m]prosrv”
  dba

strt “$DLC/bin/_mprshut $DBNAME –C biw”
  progress
  “[b]iw”
  dba
```

© copyright Éditions Eyrolles

```
strt "$DLC/bin/_mprshut $DBNAME –C aiw"
  progress
  "[a]iw"
  dba

strt "$DLC/bin/_mprshut $DBNAME –C apw"
  progress
  "[a]pw"
  dba

strt "$DLC/bin/_mprshut $DBNAME –C watchdog"
  progress
  "[w]atchdog"
  dba

msg "Creating DB_IS_UP file..."
touch $TMP/DB_IS_UP
chmod 644 $TMP/DB_IS_UP
sleep 5

# Remove user lockout flag file

if [ "$1" != "maintenance" ]; then
  msg "Users allowed into database"
  rm $TMP/User_Lockout
else
  msg "*** Database is up in maintenance mode ***"
  echo "Database is in maintenance mode – Please wait" > $TMP/User_Lockout
fi

sleep 5
msg "Database startup has completed"
```

Démarrer une session

Le démarrage d'une session peut être très simple. Placer un message dans les fichiers de verrouillage aide les utilisateurs à comprendre l'état de la base de données, quand elle n'est pas disponible. La commande **sleep 30** (secondes) donne la possibilité de lire les messages et diminue la vitesse de reconnexion à l'application. Enfin, exécuter Progress avec **exec** améliore la sécurité, en évitant qu'un utilisateur puisse revenir au shell si ce script est exécuté depuis le processus de login.

```
# Start session
#

. /mnt/scripts/env      # export TMP=/mnt/tmp

if [ -f $TMP/User_Lockout ]; then
  clear
```

© copyright Éditions Eyrolles

```
  cat $TMP/User_Lockout
  sleep 30
  exit 0
fi

# continue startup…

exec $DLC/bin/_progres –pf $SCRIPTS/standard.pf -T $TMP
```

La commande **trap** du shell est quelquefois utilisée pour empêcher l'accès au shell, mais ce n'est pas une bonne idée : Progress se sert des messages pour la communication interprocessus, et il vaut mieux ne pas interférer avec la gestion des messages, car cela peut conduire à de sérieux problèmes de fiabilité et de performance.

Arrêter la base de données

Une base de données s'arrête comme ceci :

```
# Stop server
#

. /mnt/scripts/env      # export TMP=/mnt/tmp

# Create logging directory

if [ ! -d $LOGS ]; then
  mkdir $LOGS
  chmod 755 $LOGS
fi

msg “Stopping database”
msg “Locking out users”

echo “Stopping Database – Please wait” > $TMP/User_Lockout
chmod 644 $TMP/User_Lockout
rm -f $TMP/DB_IS_UP
sleep 5

$DLC/bin/proshut $DBNAME –by >> $LOGS/stop.log 2>&1 &
sleep 60

# Verify that broker has shut down – the –by option
# Does not guarantee shutdown if there are some users in the system

ps | grep "[m]prosrv -pf $DBNAME" > /dev/null

if [ $? = "0" ]; then
       msg "*** DATABASE DID NOT SHUTDOWN WITH JUST -BY" dba
       msg "*** Attempting to force shutdown with -F option..."
```

© copyright Éditions Eyrolles

```
    $DLC/bin/proshut $DBNAME -F –by >> $LOGS/stop.log 2>&
fi
ps | grep "[m]prosrv -pf $DBNAME" > /dev/null

if [ $? = "0" ]; then      # panic
      msg "*** DATABASE DID NOT SHUTDOWN WITH -F" dba
      cp $DBNAME.lk $LOGS/$DBNAME.xxx
      exit 0
fi

msg “+++ Database is shutdown”

# sweep. lg file into archive

mv $DBNAME.lk $LOGS          # This won't happen unless the shutdown is clean…
```

Sauvegarder la base

Il y a deux approches de la sauvegarde d'une base de données Progress. On peut utiliser soit le programme **PROBKUP** fourni par Progress, soit un sous-programme de sauvegarde externe, comme **tar** ou **cpio**, ou un produit commercial. La seule manière de sauvegarder correctement une base de données en ligne est de se servir de **PROBKUP**. Une sauvegarde externe ne peut pas sauvegarder correctement une base de données Progress en service : elle doit être arrêtée au préalable.

Une excellente méthode de sauvegarde est de combiner les deux méthodes en se servant de **PROBKUP** pour faire une sauvegarde sur disque, puis d'une commande externe pour copier les fichiers disque sur bande. Si l'on dispose d'assez d'espace disque, cette approche présente plusieurs avantages :

- moins de temps d'arrêt — une sauvegarde sur disque est beaucoup plus rapide qu'une sauvegarde sur bande ;
- moins de temps de restauration — dans beaucoup de scénarios de restauration, avoir les fichiers de sauvegarde sur disque économise le temps d'obtenir les bandes depuis un emplacement extérieur au site, gain qui vient s'ajouter à la vitesse accrue des accès disque, par rapport aux accès bande ;
- une couche supplémentaire de protection — on n'a jamais trop de sauvegardes ;
- la capacité de combiner facilement les éléments non bases de données de l'application sur une seule bande.

Voici un exemple de script de sauvegarde :

© copyright Éditions Eyrolles

```
# Backup
#
. /mnt/scripts/env        # export TMP=/mnt/tmp

if [ -f $TMP/User_Lockout ]; then
  clear
  cat $TMP/User_Lockout
  sleep 30
  msg "*** Backup of active database???" dba
  exit 0
fi

# continue backup…

msg "*** Starting PROBKUP of $DBNAME"

$DLC/probkup $DBNAME /mnt/dump/$DBNAME.bak >> $LOGS/backup.log 2>&1

msg "*** PROBKUP of $DBNAME complete with status $?"

tar cvf /mnt/dump/$DBNAME.bak >> $LOGS/backup.log 2>&1

msg "*** Backup of $DBNAME complete with status $ ?"
```

Il est essentiel de tester périodiquement les procédures de restauration pour s'assurer de la validité de la sauvegarde et de l'exhaustivité du contenu des bandes. La meilleure méthode est de restaurer sur une machine différente. Cela vérifie l'ensemble du processus et tous les média. Le PEG (*Progress Email Group*, www.peg.com) raconte de temps à autre l'histoire triste d'âmes en peine, qui pensaient avoir fait des sauvegardes toutes les nuits, mais d'un coup découvraient qu'ils n'avaient plus rien. Evitez de vous joindre au club, et vérifiez votre sauvegarde.

Surveillance

Progress propose un excellent moniteur de base de données, en mode caractère, connu sous le nom de **PROMON**. Les administrateurs doivent se familiariser avec **PROMON** et son mode interactif et, par ailleurs, s'accoutumer à l'ordre d'apparition des valeurs pour leur système. Mais la vraie puissance de cet outil réside dans la constitution d'une synthèse historique de l'activité sur une longue période de temps. On peut alors utiliser ces données pour valider une hypothèse concernant un problème et décider d'une action de manière beaucoup plus ciblée et fiable. Le script suivant peut servir de base pour rassembler les données :

```
# Monitor
#

. /mnt/scripts/env        # export TMP=/mnt/tmp
TM=`date "%H.%M"`
```

© copyright Éditions Eyrolles

```
$DLC/bin/promon $DBNAME > $LOGS/sample.$TM <<- "EOF" 2> /dev/null
   R&D
   5
   3
   300
   1
   9999
   p

   2
   1
   s
   p
   x
EOF
```

Ce script démarre **PROMON**, choisit le menu R&D (plus détaillé que le menu par défaut), et règle l'intervalle d'échantillonnage à 300 secondes et le nombre de lignes par page de la sortie à 9999. Il retourne au menu précédent, va à l'écran de résumé d'activité, récupère le résumé depuis le démarrage de la base de données, échantillonne l'activité pendant cinq minutes et sort.

```
01/18/00 Activity : Summary
15:04:35 01/18/00 14:59 to 01/18/00 15:04 (5 min 1 sec)

Event               Total   Per Sec   |Event            Total   Per Sec

Commits             17951      59.6   |DB Reads         26059      86.5
Undos                   2       0.0   |DB Writes          903       3.0
Record Reads      1219453    4051.3   |BI Reads             0       0.0
Record Updates       5144      17.0   |BI Writes          554       1.8
Record Creates       9045      30.0   |AI Writes          316       1.0
Record Deletes        869       2.8   |Checkpoints          0       0.0
Record Locks        77537     257.5   |Flushed at chkpt     0       0.0
Record Waits            0       0.0

Rec Lock Waits      0 %  BI Buf Waits      0 %   AI Buf Waits      0 %
Writes by APW     100 %  Writes by BIW    42 %   Writes by AIW    75 %
DB Size:     95384136 K  BI Size:     255984 K   AI Size:      72656 K
Empty blocks:  1002347  Free blocks:        0   RM chain:         8 .
Buffer Hits        99 %  Active trans:      13

3 Servers, 679 Clients (679 Local, 0 Remote, 11 Batch), 2 Apws
```

Exécuter ce script depuis **cron** à intervalles réguliers (toutes les demi-heures est un bon début) : cela constitue un échantillon pour surveiller et contrôler la base de données. Il existe un grand choix d'autres informations intéressantes dans **PROMON** — ajoutez-en autant que vous le pouvez. Il est utile aussi de rassembler les informations système qui coïncident avec ces données pour analyser les corrélations possibles. Par exemple :

© copyright Éditions Eyrolles

```
cd /mnt/tmp

ls -l DBI* > $LOGS/dbi.$TM
ls -l srt* > $LOGS/srt.$TM
ls -l lbi* > $LOGS/lbi.$TM

cd /mnt

ls -l db bi ai > $LOGS/dblist.$TM

df > $LOGS/df.$TM
who –u > $LOGS/who.$TM
ps aux > $LOGS/ps.$TM
netstat -i > $LOGS/netstat.$TM
vmstat > $LOGS/vmstat.$TM
top –b –n1 > $LOGS/top.$TM
```

Les données capturées sont susceptibles de fournir des indications très importantes concernant l'état du système dans son entier pendant la période durant laquelle un échantillonnage particulier s'est effectué.

La surveillance de la base de données et du système d'exploitation n'est qu'une partie du combat — peut-être la plus importante et celle qui offre le plus d'opportunités pour surveiller l'application elle-même. La complainte de l'utilisateur, « c'est lent », est sûrement l'un des aspects les plus frustrants du travail de l'administrateur. « Qu'est-ce qui est lent ? » et « Comment est-ce lent ? » sont les deux questions clés, auxquelles il est souvent très difficile de répondre. Une technique simple pour glaner quelques idées de réponses est d'exécuter une session, en tâche de fond, qui mesure les temps de réponse et rassemble les statistiques d'utilisation de l'application. Cela nécessite une petite procédure en L4G :

```
/* zippy.p */

define variable l as character no-undo.
Define variable stime as integer no-undo.
Define variable endtime as integer no-undo.
Define variable ttime as integer no-undo format ">>>>>9".
Define variable utime as integer no-undo format ">>>>>9".
Define variable rcount as integer no-undo.
Define variable nxtapp as integer no-undo initial 100000.
Define variable z as integer no-undo.
Define variable next_time as integer no-undo initial 21600.  /* 6am */
Define variable curord as integer no-undo.
Define variable oldord as integer no-undo format ">>>>>9".
Define variable l as integer no-undo.
Define variable r as integer no-undo extent 100.
Define variable w as integer no-undo extent 5.
Define variable t as integer no-undo.
```

© copyright Éditions Eyrolles

```
Define variable rt as integer no-undo.
Define variable wt as integer no-undo.
Define variable mr as integer no-undo.
Define variable mw as integer no-undo.
Define variable fname as character no-undo.
Fname = "/mnt/logs/" + string( month( today ), "99" ) + "." +
        string( day( today ), "99" ) + "/zippy.dat".

do while true.
  Pause 30.
  If time < next_time then next.
  Next_time = 300 * integer( time / 300 ).
  Next_time = next_time + 300.    /* increment time by 5 minutes.   */

  /* 100 sort of random record reads of 5 common tables (20 each) */
  r = 0.
  Stime = etime( yes ).

  Do rcount = 1 to 20:
    Stime = etime.
    Find next customer no-lock no-wait no-error.
    Find next order no-lock no-wait no-error.
    Find next order-line of order no-lock no-wait no-error.
    Find next salesrep no-lock no-wait no-error.
    Find next state no-lock no-wait no-error.
  End.

  Ttime = etime.
  L = " ".
  W = 0.
  Utime = etime( yes ).

  /* 5 writes */
  do I = 1 to 5:
    utime = etime.
    Do for state TRANSACTION:
      find first state exclusive-lock no-wait.
      if available state then
        state.state = state.state.
       else
        I = "x".
    end.
    w[i] = etime – utime.
  End.

  Utime = etime.
  Curord = current-value( next-ord-num ).
  Oldord = ( if oldord = 0 then 0 else ( curord – oldord )).
  T = 0.
  Mw = 0.
```

© copyright Éditions Eyrolles

```
  Do I = 1 to 5:
    t = t + w[i].
    Mw = max( mw, w[I] ).
  End.
  Wt = t / 5.

  T = 0.
  Mr = 0.
  Do I = 1 to 100:
    t = t + r[i].
    Mr = max( mr, r[I] ).
  End.
  Rt = t / 100.

  Output to value( fname ) append.
  Put string( time, "HH:MM" ) + " " + I.
  Put
    Ttime
    Utime
    Oldord
    " ".
  export mr rt mw wt.
  Output close.

  Oldord = curord.
End.
Return.
```

Ce programme fonctionne en arrière-plan, connecté à la base de données **sports.** Toutes les cinq minutes, il lit 100 enregistrements pseudoaléatoires, écrit un enregistrement (qui ne doit jamais être modifié ou verrouillé) et calcule le nombre de nouvelles commandes reçues dans l'intervalle. Les statistiques et le nombre de commandes sont sortis toutes les cinq minutes. Vous pouvez surveiller la sortie à l'aide de :

```
$ tail –f /mnt/log/01.12/zippy.dat
```

Les pointes, dans les résultats, indiquent des événements intéressants — si le nombre de commandes présente des pointes, c'est que les données sont actives, et on peut admettre que le temps de réponse en lecture et en écriture augmente légèrement. De grandes variations dans le temps de réponse, avec ou sans variations correspondantes de l'activité, tendent à indiquer qu'il y a un problème dans l'environnement, qu'il va falloir identifier — cela peut signifier, par exemple, qu'une autre application, qui partage les disques, les surchargent trop au point que l'application Progress en est affectée, ou bien que l'on fonctionne trop près de la limite de capacité de la configuration courante et qu'un réglage ou une mise à niveau sont à prévoir.

© copyright Éditions Eyrolles

On a intérêt à rassembler des données relatives à la performance, qu'il s'agisse de mesures ou d'indicateurs pertinents de l'activité (tels que le nombre de commandes) et à les comparer : c'est une aide précieuse au moment de définir une mise à niveau du système, ou d'examiner d'autres solutions — s'il n'y a aucune augmentation de l'activité, pourquoi des problèmes de performance apparaissent-ils ? Il y a sûrement quelque chose de mal configuré ou de mal conçu quelque part.

Dépannage

Il y a peu à redire sur une base de données Progress. La plupart des problèmes se produisent sans doute du fait d'utilisateurs forcenés. Voici quelques-uns des scénarios les plus probables.

Utilisateurs suspendus, emballements et autres psychodrames

Le moyen le plus simple (et le plus sûr) de gérer les processus **_progress** (sessions utilisateur en self-service), qui demande quand même une intervention externe, est d'attendre qu'ils soient terminés. Si ce n'est pas la solution, alors il faudra, dans ce cas, écarter le plus gentiment possible l'utilisateur de son clavier, tout en gardant un œil sur ses mains. Ce qui évitera (on l'espère) tout problème qui pourrait surgir de tentatives de remédier simultanément à la situation. Le meilleur parti est **kill -2** (**SIGINT**), qui active la condition **STOP** dans les procédures L4G et relance la procédure de démarrage **-p**. La tâche n'est pas détruite, mais il ne se passe rien de mauvais. Et si le problème est survenu pendant la « suspension », l'utilisateur pourra probablement revenir dans sa session sans avoir à faire un nouveau login UNIX (si, par contre, c'est un problème réseau ou de PC, vous n'avez pas de chance).

Si **kill -2** ne suffit pas — par exemple, si l'on a limité à un le nombre de login d'un utilisateur et qu'il y a un problème réseau, la session au repos bloque l'accès — alors on utilisera encore **kill -2** pour stabiliser le processus. Une fois qu'il est stabilisé (le fichier **.lg** dit que toute transaction est achevée, qu'il ne consomme plus aucun temps du processeur et n'exécute plus d'opérations d'entrée/sortie) **kill -1** (**SIGHUP**) ou le terminera de manière très sûre, comme si l'utilisateur avait éteint son terminal. **kill -15** (**SIGTERM**) a le même effet.

Ce n'est pas une bonne idée de se servir d'une série de commandes **kill** sauf si l'on peut les espacer dans le temps, et si l'on contrôle à la fois le fichier **.lg** et l'état du processus. Si l'on interrompt les sous-programmes Progress d'interruption propre au profit d'une interruption non gérée correctement, on peut facilement aboutir à une panne totale de la base de données.

© copyright Éditions Eyrolles

La capture des messages par la commande **trap** du shell n'est pas une bonne initiative : elle met en défaut le fonctionnement de ces messages et conduit à un état très instable du système. Si l'on sort d'une session _**progress**, que l'on tape **trap** et que l'on obtient une sortie quelconque, alors le système est en état de risque. Le problème le plus courant dans ce scénario est un processus d'emballement (*runaway*) — un processus qui tente de consommer tout le temps CPU disponible. S'il y a un **trap**, il faut s'en débarrasser.

Le comportement de **kill -8** (**SIGFPE**) varie selon les versions de Progress. Dans certaines versions, il agit comme un message non géré — c'est-à-dire qu'il vide la mémoire et que le processus meurt sans tentative sérieuse de nettoyage, ce qui revient, pour l'essentiel, à la même chose que **kill -9**. C'est mauvais : si l'on a laissé un verrou, cela peut entraîner l'arrêt de la base de données. Dans les versions les plus récentes (dont la version 8.3b de Linux), cette commande se comporte comme si **SIGHUP** avait été envoyé, mais elle crée aussi un fichier **procore.**

Avec **kill –10** (**SIGUSR1**), dans les versions récentes, le processus crée un fichier **procore** et un fichier **protrace,** sans autre impact sur le processus (qui reste en fonctionnement). C'est une caractéristique « super » bien qu'elle ne soit pas encore très bien prise en charge (elle n'est documentée nulle part et le support technique ignore souvent son existence). Le fichier **protrace** est très intéressant — il contient une trace de la pile L4G, dont le numéro de ligne indiquant où en était le processus au moment où il a été « signalé ». C'est très pratique pour dépister ce qu'un utilisateur faisait (vraiment) quand il a été « suspendu ».

kill –9 (**SIGKILL**) n'est jamais bon et n'est jamais vraiment nécessaire. Il existe un mythe selon lequel **kill –9** est « bon » parce que cela « marche toujours ». C'est faux — un processus bloqué dans certains états (tels que **D** ou disk-wait) ne réagira à rien, y compris à **kill –9**.

Quand on utilise **kill –9**, le processus détruit ne peut pas être nettoyé (c'est une caractéristique UNIX — **kill –9** est défini comme *uncatchable*, impossible à rattraper). Le chien de garde (*watchdog*) remarque que le client est parti et tente alors d'arrêter la session et de nettoyer tout ce qui est actif sous le compte du client. Il finit par nettoyer la plupart du temps. Quelquefois une ressource critique est verrouillée, et il n'y a aucun moyen de nettoyer parce que les données nécessaires qui étaient dans l'espace mémoire du client ont disparu. Face à cette éventualité, le chien de garde arrête la base de données — si l'on n'utilise pas de *watchdog*, l'agent (*broker*) le fera. Ce problème n'est pas aussi important qu'il en a l'air : si un verrou a été maintenu (il y en a toujours quand le chien de garde arrête brutalement la base de données), personne n'a rien fait d'autre qu'essayer de contourner le problème, de toute façon, les utilisateurs sont bloqués, et ils ne le savent pas encore.

© copyright Éditions Eyrolles

La corruption, que beaucoup ne constatent qu'après s'être servis de **kill -9**, provient d'autres événements concomitants. En aucun cas, l'écriture de « mauvaises données » sur le disque ne peut résulter directement d'une commande **kill**, qu'il s'agisse d'un fichier **.bi** ou du fichier **.db**. Des actions inconsidérées telles que la suppression des fichiers **.lk** vivants, la modification de la localisation du fichier **.bi,** l'utilisation imprudente de **dbrpr**, ou l'usage de l'option **-F** *peuvent* produire une corruption des données (il faut quand même ignorer bon nombre de messages d'avertissement pour y parvenir). Les environnements où **kill -9** est utilisé sont souvent très peu contrôlés et chaotiques. Beaucoup de bêtises sont commises, mais ne font pas l'objet d'un rapport ou d'une réflexion, de sorte qu'il est difficile d'aller au fond des choses. Et c'est **kill -9** qui se retrouve incriminé.

En résumé, **kill -9** sera, un jour ou l'autre, cause d'une panne de la base de données — c'est comme jouer à la roulette russe. Il ne produit pas, en soi, de corruption de la base de données. Cependant, il crée les conditions d'autres activités qui sont susceptibles d'endommager les données. Si vous estimez devoir arrêter un processus, utiliser d'abord **kill -2**, puis, si nécessaire, **kill -1**. Si cela ne suffit pas, alors quelque chose d'autre va vraiment mal et vous avez besoin d'aide.

Forces et faiblesses

Comme tout produit, Progress n'est pas parfait. Bien qu'il offre un vaste ensemble de caractéristiques, que seules possèdent les bases de données de classe professionnelle, il lui manque certaines des options les plus chères disponibles chez les « quatre grands ». Voici un exposé de ses forces et de ses faiblesses.

Le L4G

Pour beaucoup de raisons, le L4G (langage de quatrième génération) de Progress est à la fois sa plus grande force et une faiblesse. Le L4G est un langage procédural conçu pour écrire une application entière. Depuis le tout début, Progress Software Corporation (PSC) a tenu cette promesse, comme en témoignent les nombreuses applications de ses partenaires, qui ne font guère appel au L3G (langage de troisième génération). Il ne s'agit pas seulement de produits marginaux, mais des produits industrialisés des éditeurs de logiciel tels que FiServ, qad Inc., Symix et NxTrend. D'un autre côté, les L4G sont perçus comme propriétaires et fermés. Progress ne fait pas exception et, en fait, le manque de pilotes ODBC abordables a jusqu'à récemment renforcé le caractère fermé du produit. À un certain niveau, c'est un faux problème

© copyright Éditions Eyrolles

puisqu'en fait toutes les bases de données ont leurs propres extensions SQL propriétaires et que la portabilité du code est limitée à un petit sous-ensemble de l'application.

Orientation OLTP

Depuis le début, le moteur de base de données a été construit comme un moteur de transactions. En fait, contrairement aux autres bases de données, il est impossible de modifier des données sans passer par une transaction — il en résulte une très forte intégrité des données. Le L4G lui-même a des caractéristiques orientées vers le monde OLTP. Traditionnellement, SQL s'appuie sur les ensembles, ce qui signifie qu'à un instant donné, les enregistrements sont regroupés dans un ensemble, avant d'être traités. Pour les applications en ligne, ce n'est pas aussi efficace et naturel que le L4G qui s'attache à trouver un enregistrement unique et le rend disponible à l'application immédiatement.

Cependant, il y a un revers au modèle OLTP pur, c'est l'aide à la décision. Les accès orientés ensemble sont de loin plus performants que les recherches individuelles d'enregistrements quand il s'agit de statistiques et de prévisions à grande échelle. Les générateurs de rapports ODBC attendent de telles caractéristiques une rapidité au moins égale à celle de la vitesse d'exécution du SQL. Progress supporte la spécification SQL-88, la plus ancienne, mais il n'est pas optimisé pour ce type d'accès. Les rapports fonctionnent plutôt bien mais, dans beaucoup de cas, sont inefficaces et écrits en mauvais SQL : le SQL généré par ordinateur, en particulier, ne fonctionne pas aussi vite que le L4G natif pour les même rapports.

Toutefois, cette situation est en train de changer. La version 9 introduit lentement le moteur SQL-92, récemment écrit. PSC a écrit un produit entièrement nouveau qui s'appuie sur le même gestionnaire de mémoire que le produit L4G courant. Le nouveau produit ne se contente pas seulement de se conformer au standard de langage le plus récent, il est optimisé pour exécuter SQL. Ce nouveau produit sera commercialisé avec la version 9.1 de Progress.

Restauration après panne « à l'épreuve des balles »

Parmi les forces de ventes de Progress, circule une vieille histoire sur la manière dont les représentants étaient entraînés à faire la preuve de la fiabilité de la base de données. Des démonstrations étaient faites sur des ordinateurs portables, fonctionnant sous MS-DOS. Il y a toujours eu un problème sur cette plate-forme vis-à-vis des fichiers ouverts, en cas de panne : c'était connu, ils en prenaient pour leur grade. Au milieu de la démonstration, le représentant devait arracher du mur le cordon d'alimentation de la machine qui, sans soutien d'une batterie, devait tomber en panne. Sans exception, la base de données était remise en état et restaurée avec succès après la panne.

© copyright Éditions Eyrolles

La technologie qui entre dans le traitement des transactions de Progress est la même que celle qui préserve la base de données de la panne d'alimentation ravageuse. Il ne s'agit pas d'une panacée en cas de panne matérielle. Toutes les bases de données perdront des données si le média physique est irrécupérable au niveau du matériel.

Coût de propriété

Progress est proposé à un prix abordable pour les sociétés ordinaires — pas seulement en termes d'investissement initial, mais aussi en termes de coût de fonctionnement. Aujourd'hui, la plupart des installations Progress n'ont pas d'administrateur de base de données à temps plein. La base de données n'a besoin d'être reconfigurée que dans très peu de cas, tels qu'une augmentation significative du nombre d'utilisateurs. La plupart des administrateurs Progress partagent leur temps entre une heure ou deux de surveillance de leur base de données et d'autres tâches banales d'administration du système. Un système qui peut convenir à plusieurs milliers d'utilisateurs tout en étant opérationnel avec une administration de base de données minimale est un remarquable succès.

PSC a autorisé le groupe Aberdeen à évaluer les coûts initiaux et les coûts à long terme de la base de données Progress, comparée aux bases de données qui ciblent les environnements grands et petits. Les résultats peuvent être consultés sur le site web de Progress, à http://www.peg.com.

Indexation des mots

L'une des perles souvent oubliée de la base de données Progress est une fonctionnalité connue sous le nom de *Word index*. Les données de type caractère servent à constituer un index qui identifie tous les mots contenus dans les données. Ce n'est pas une caractéristique particulièrement sophistiquée de recherche de texte, mais c'est un moyen peu coûteux d'opérer une recherche avancée sur les bases de données de messages, qu'il s'agisse de groupes de news, d'informations pratiques ou de bases de connaissances. L'accès à l'index est fourni par la simple fonction **CONTAINS** du L4G.

Internationalisation

L'internationalisation couvre une large variété de sujets, tels la version Double Byte de Progress destinée aux jeux de caractères asiatiques, la traduction des messages d'erreur de Progress, la date régionalisée et les formats numériques, ou les tables de fusion fournissant des capacités de tri qui reposent sur les ensembles de caractères nationaux. Progress met tout cela à disposition au travers d'un riche ensemble de fonctionnalités de gestion de caractères.

© copyright Éditions Eyrolles

De plus, Progress propose un gestionnaire de traduction, en option. Sa fonction principale est de parcourir le code source du L4G Progress pour trouver les chaînes utilisées dans l'application en tant que messages et légendes d'écran. Cette liste peut alors être confiée à un traducteur professionnel qui n'a pas besoin de connaître l'informatique. Les copies des chaînes traduites peuvent alors être conservées ou faire l'objet d'un remplacement par le gestionnaire de traduction.

Fonctionnement 24 heures sur 24

Progress a ajouté une fonctionnalité de sauvegarde en ligne dans plusieurs des versions précédentes ; mais il manque encore quelques caractéristiques au produit actuellement disponible pour Linux. La version 8.3 ne permet pas de réorganiser les index quand la base de données est en ligne. L'activation et la désactivation des index, ainsi que la plupart des modifications du schéma de la base de données, ne peuvent actuellement pas être effectuées en ligne.

Progress a, petit à petit, ajouté des caractéristiques pour atténuer quelques-uns de ces problèmes. Par exemple, de nouveaux champs peuvent maintenant être ajoutés, sans qu'il faille réorganiser toute la table. De plus, la version 9 offre maintenant le transfert de table et la construction d'index en ligne. La plate-forme Linux devrait en bénéficier avec l'introduction de Progress 9.1.

Aucun BLOB

Les grands objets binaires (BLOB, *Binary Large OBjects*) sont des données telles que les clips vidéo et audio, ou les données géospatiales. La version courante de Progress ne gère pas ces données, mais un support minimal est prévu à l'avenir. Les objets plus petits, jusqu'à une taille de 32 Ko, sont gérés ; on peut aussi se servir de la base de données comme d'un index de BLOB, rangés en tant que pointeurs de fichiers. Cependant, si l'on a des contraintes de gestion d'objets multimédia très importants, Progress n'est probablement pas la bonne solution.

Pas de requêtes parallèles

Seules un petit nombre de bases de données de haut de gamme ont incorporé la capacité de traiter une requête de base de données simple à l'aide de processeurs multiples. La technologie sous-jacente a montré qu'elle est extrêmement coûteuse et demande du temps de débogage. Les architectures à processeurs massivement parallèles actuelles peuvent certainement fournir d'immenses ressources, mais le coût de telles machines et le coût de création de la technologie les placent hors de portée des départements informatiques équipés de matériels de taille moyenne. De plus, il n'y a aucune garantie que les requêtes individuelles bénéficieront du traitement sur plusieurs proces-

© copyright Éditions Eyrolles

seurs. Souvent, on s'aperçoit que les opérations d'entrées/sorties deviennent le goulot d'étranglement, à la place du processeur et, dans ce cas, tout l'argent dépensé sur un système de requêtes parallèles n'aura servi à rien.

Aucune gestion de verrouillage distribué

Pour fournir un rollback transactionnel complet entre bases de données multiples, une technique dite commit à deux phases est nécessaire. Elle fit son apparition peu après l'introduction de la possibilité, pour une même session Progress, de se connecter simultanément à plusieurs bases de données. Comme le schéma de commit à deux phases implique des données partagées en mémoire principale, il n'y a aucun problème si ces bases de données sont implantées sur le même système. Mais si les bases de données sont situées sur des machines différentes, la méthode ne fonctionne plus. Pour disposer d'une telle caractéristique, il faut un gestionnaire de verrouillage distribué. La théorie de ce type de logiciel est complexe et la pratique coûteuse : c'est de plus en plus cher à mesure que l'on s'en sert. PSC a pris position : le coût d'un tel gestionnaire de verrouillage sera beaucoup trop élevé pour un client de base soucieux d'utiliser une base de données de haute rentabilité (ROI). Par conséquent, les systèmes qui demandent des produits comme Tuxedo et TopEnd ne sont pas des candidats pour Progress.

Une interface CGI simple vers Progress

Le serveur http standard de processus CGI se contente de fournir **stdin** et **stdout** à un programme, programme qui peut être n'importe quoi. Le script de shell suivant passe tout simplement le **stdin** à une pile FIFO classique qui, cela tombe bien, est justement lue, en tâche de fond, par une procédure Progress L4G (ce pourquoi elle est classique… les procédures étaient d'avance bien rôdées). Puis il lit une réponse d'une FIFO qu'il a créée pour les besoins du serveur Progress. Il faut ici s'assurer que la réponse est bien retournée à l'utilisateur — voir le chapitre 3 du *Guide des interfaces externes Progress* pour plus d'informations. Le programme Progress reçoit les requêtes et y répond — la formulation, dans notre exemple, de la demande et de la réponse sont, évidemment, peu intéressantes mais on les modifie facilement :

```
# !/bin/sh
#
# cgi-bin/xxx

PIPE=/tmp/cgi$$          # l'endroit où envoyer la réponse
rm -f $PIPE              # s'assurer qu'il n'y a rien en suspens
/etc/mknod $PIPE p       # créer le canal nommé
```

© copyright Éditions Eyrolles

```
chmod 666 $PIPE        # permettre l'ecriture
read INPUT             # lire l'entrée

# ajouter le nom du canal nommé pour envoyer la réponse au début du formulaire
# des données et écrire tout le message au canal "bien connu" /tmp/cgi-pipe
echo "$PIPE&"$INPUT > /tmp/cgi-pipe

cat < $PIPE            # lit la réponse et la renvoie au serveur
rm -f $PIPE            # nettoie
```

Voici la partie en L4G Progress :

```
/* websrvr.p */
define variable I as integer no-undo.
define variable req as character no-undo.
define variable outpipe as character no-undo.

/* enlever ce commentaire pour que le démon progress crée le canal...
unix silent value( "rm -f /tmp/cgi-pipe" ).
unix silent value( "/etc/mknod /tmp/cgi-pipe p" ).
*/

do while true :
  input from value( "/tmp/cgi-pipe" ) no-echo.
  import unformatted req.
  outpipe = substring( req, 1, index( req, "&" ) - 1 ).
  output to value( outpipe ) unbuffered.
  put unformatted "Content-type : text/html" skip(1).
  put unformatted skip(1).

  /* faire quelque chose comme analyser la requête, chercher les données et
     envoyer la réponse...
   */

  I = I + 1.
  put unformatted
    "<html><head><title>Bogus Response</title></head></html>" +
    "<body><p>" +
    "Sample Bogus Response #" + string( I, ">>>>9" ) +
    "</p></body></html>".
    output close.
    input close.
  end.
return.
```

Le HTML correspondant est le suivant :

```
<form action="cgi-bin/xxx" method="post">
What ? <input name="something" size="30" maxlength="30">
<input type="submit" value="submit">
</form>
```

© copyright Éditions Eyrolles

Un script très simple lance le démon :

```
# Start daemon
#
. /mnt/scripts/env        # export TMP=/mnt/tmp

if [ -f $TMP/User_Lockout ] ; then
  clear
  cat $TMP/User_Lockout
  sleep 30
  exit 0
fi
# startup…

sh -c "eval nohup $DLC/bin/_progres -b –p websrvr.p >> $LOG 2>&1 &"
```

Tout cela représente un moyen simple et direct d'accéder à Progress *via* une interface web. Évidemment il y a des problèmes de sécurité et d'échelle avec cette méthode, mais elle se prête bien à un usage interne rapide et trivial, car elle est très puissante dans ce cas.

Voici un exemple de table HTML simple, construite à partir de données de la base **sports** fournie avec le produit. Elle contient une liste d'informations sur les clients :

```
/* sportssrvr.p */

define variable I as integer no-undo.
define variable req as character no-undo.
define variable outpipe as character no-undo.

/* enlever ce commentaire pour que le démon progress crée le canal...
unix silent value( "rm -f /tmp/cgi-pipe" ).
unix silent value( "/etc/mknod /tmp/cgi-pipe p" ).
*/

do while true :

  input from value( "/tmp/cgi-pipe" ) no-echo.
  import unformatted req.
  outpipe = substring( req, 1, index( req, "&" ) - 1 ).
  output to value( outpipe ) unbuffered.
  put unformatted "Content-type : text/html" skip(1).
  put unformatted skip(1).

 /* analyser la requête, chercher les données et envoyer la réponse...
 */
 I = I + 1.
```

© copyright Éditions Eyrolles

```
put unformatted
   "<html><head><title>Customer List</title></head></html>" +
   "<body><br><br><B><I>Customers :</B></I><br>"
   "<table><th>Number</th><th>Name</th><th>Address</th>"
   "<th>City</th><th>State</th><th>Zip</th><th>Country></th>".
 for each customer no-lock :
    put unformatted "<tr>"
                    "<td>" customer.cust-num "</td>"
                    "<td>" customer.name "</td>"
                    "<td>" customer.address "</td>"
                    "<td>" customer.city "</td>"
                    "<td>" customer.state "</td>"
                    "<td>" customer.postal-code "</td>"
                    "<td>" customer.country "</td>"
                    "</tr>".
 end.

 put unformatted "</table></body></html>".
   output close.
   input close.

 end.

return.
```

Résumé

La base de données Progress prouve depuis des années qu'elle est un produit extrêmement fiable, aussi facile à utiliser qu'à administrer. L'installation représente à peine plus qu'une copie du média et l'initialisation de quelques variables. Bien qu'il y ait beaucoup de caractéristiques telles que la restauration avancée (*roll forward*) et le commit à deux phases, il n'y a rien d'obligatoire et l'administration est simple. Les nombreuses options de démarrage permettent de personnaliser le fonctionnement de Progress, d'augmenter les performances et d'équilibrer l'utilisation des ressources. Bien que Progress soit un excellent choix pour les PME sensibles aux coûts, certaines grandes entreprises peuvent exiger des caractéristiques de fonctionnement 24 heures sur 24, non disponibles dans la version 8.3 fournie sur Linux. En dehors de ces caractéristiques de haut de gamme, Progress fournit toutes les caractéristiques vraiment importantes que les bases de données de renom proposent à un prix beaucoup plus élevé.

© copyright Éditions Eyrolles

FAQ

Q. Quel est l'âge de Progress Software Corporation (PSC) ?

R. Au début, PSC a démarré sous le nom de Data Language Corp., en 1981. C'est l'origine du nom de la variable DLC utilisée lors de l'installation. Le produit s'appelle Progress depuis sa commercialisation, en 1983. Pour bénéficier de cette image, le nom de la compagnie a changé en 1987. PSC est cotée depuis 1991 et connue au Nasdaq sous l'appellation de PRGS.

Q. Quelle est la différence entre les versions Workgroup et Enterprise de la base de données ?

R. Workgroup a été conçue pour être utilisée par un maximum de 35 utilisateurs, et strictement 50 utilisateurs. La plupart des caractéristiques de performance haut de gamme, tels les blocs de base de données variables, l'écriture de pages de background ou le paramètre –**spin**, ne sont pas disponibles. Toutes les caractéristiques de base, restauration avancée, tolérance aux pannes, mise en réseau, connexions de base de données multiples dans une connexion simple et commit à deux phases sont communes aux deux bases de données. Mais les utilisateurs, quels que soient leurs besoins, ont intérêt à ne pas négliger, le cas échéant, le choix Enterprise, si des performances sont attendues.

Q. Progress est-il disponible à l'international ?

R. Oui. En fait, Progress a des bureaux partout dans le monde. Dans certains cas où Progress n'est pas présent, des réseaux de distribution nationaux ont été créés. Chaque bureau a sa propre politique et ses tarifs, aussi n'ayez pas peur de vous adresser à votre bureau local. La liste est disponible sur le site web de Progress.

Q. Y a-t-il un niveau RAID recommandé avec Progress ?

R. On ne recommande aucun RAID avec parité (2, 3, 4, 5, 7, …). Les résultats des RAID avec parité sont une parodie de performance. Cela n'est fait que pour vous rendre fidèle à un éditeur de RAID, car vous aurez en permanence besoin d'amélioration et de mise à niveau pour tirer des performances raisonnables d'un tel système. RAID 0 (*striping*) et RAID 1 (*mirroring*) fonctionnent sans parité et sont, par conséquent, de bons choix (en particulier s'ils sont combinés). Vous devez ajouter de la RAM et mentir en quelque sorte sur les résultats de demandes d'entrées/sorties (cache « écriture après coup » ou cache « écriture au fur et à mesure ») pour améliorer les performances des RAID avec parité, alors que l'on améliore les performances des systèmes agrégé par bande (*striped*) et en miroir en ajoutant des disques. Le disque est moins cher que la RAM (en particulier la RAM exigée pour les RAID et les serveurs à haut débit) et vous pouvez en ajouter beaucoup plus (il y a toujours des limites strictes

© copyright Éditions Eyrolles

sur le volume de RAM que l'on peut ajouter, sans même tenir compte du budget...). Et il y a aussi la question des conséquences dans la RAM en cas de panne d'alimentation. Si une écriture du cache est autorisée (il faut qu'elle le soit, pour obtenir des performances décentes, si on laisse de côté la parodie concernant les RAID), une seule petite erreur et votre base de données est nettoyée (sachant que les batteries de sauvegarde sont souvent en défaut) parce que la zone a « menti » à Progress, c'est-à-dire n'a pas montré ce qui est vraiment sur le disque, ce qui rend la restauration après la panne très contestable.

Dire **Non** à RAID 5 (et à ses cousins).

Q. Pourquoi Progress ne fonctionne-t-il pas avec RAID 5 ?

R. Progress fonctionne avec RAID 5. Mais, comme toute autre base de données, il fonctionne beaucoup, mais vraiment beaucoup plus vite avec d'autres technologies de disque. Progress n'est pas au courant du type de système RAID que vous utilisez. Seuls les utilisateurs auront une opinion.

Q. Je me sers de la version courante de Progress qui utilise les binaires SCO et iBCS. Que dois-je faire pour passer à la version Linux native ?

R. D'abord, appeler vos commerciaux PSC et modifier votre licence pour la plate-forme Linux, puis installer les binaires natifs comme décrit dans ce chapitre. Vérifiez que vous avez choisi un répertoire à part pour cette version de Progress. Modifiez vos variables d'environnement, en particulier la variable **$DLC**, et le reste de vos scripts existants de démarrage, d'arrêt, de sauvegarde, etc. seront prêts à fonctionner. La seule chose nécessaire est de convertir la base de données au format de la version 8.3 pour Linux. Si vous utilisez déjà le format 8.3 pour SCO, il n'y a pas lieu de faire la conversion. Avec la version 8.2, il s'agit seulement de tronquer le fichier **.bi,** par la commande **proutil dbname –C truncate bi**. Cependant, ces deux méthodes ne sont pas garanties par Progress. Pour des versions plus anciennes de Progress, ou par sécurité, créer une nouvelle base de données version 8.3 avec le binaire Linux, puis effectuer un déchargement et un rechargement de la base de données.

Q. Je sais que le L4G Progress dispose d'une interface graphique sous MS Windows et que Linux admet Progress avec un client en mode caractère. Mais puis-je utiliser aussi X Window ?

R. Oui, mais seulement avec l'ancienne version 7 de Progress, qui n'est pas un portage natif sous Linux (en d'autres termes, les binaires SCO se servant de l'émulation iBCS). Au moment où la version 8 était en chantier, l'équipe marketing a révisé le nombre de licences vendues en environnement graphique sous UNIX et a trouvé ce nombre un peu restreint. Une décision marketing a été de geler le support X Window au niveau de la version 7 de Progress. C'était aussi l'époque où X Window signifiait un

© copyright Éditions Eyrolles

matériel X-Terminal particulier ou une émulation coûteuse de X Window sur PC. La popularité de Linux n'était pas ce qu'elle est aujourd'hui, et le nombre d'ordinateurs capables de fonctionner avec X Window a crû dans les mêmes proportions depuis. De toutes façons, appelez votre commercial Progress et exprimez votre intérêt pour la révision du support X Window.

Q. J'ai entendu dire que l'image « après » est un gouffre de performance. Est-ce que j'en ai vraiment besoin ? Après tout, je peux toujours sauvegarder la base de données.

R. L'image « après » vous sauvera un jour ou l'autre. Elle permet de restaurer une base de données dans l'état où elle était juste une minute avant la panne. Même si vous sauvegardez le système entier toutes les nuits, vous prenez le risque d'une panne en fin de journée, et vous perdez un jour entier de travail si vous n'utilisez pas l'image « après ». Se poser toujours la question, « Puis-je me permettre de perdre des données depuis la dernière sauvegarde ? ». Sur les systèmes qui fonctionnent 24 heures sur 24 et autres systèmes critiques, la perte de données n'est, en général, pas admissible.

Q. La documentation Progress dit que je peux administrer la base de données à l'aide du dictionnaire, même si je n'ai que des clients runtime. Comment puis-je accéder au dictionnaire ?

R. Le code du dictionnaire est crypté. Démarrer simplement votre session Progress avec le paramètre **–rx**, pour indiquer que le mode source est crypté.

Q. Quand je démarre ma session Progress, je reçois le message « Cette version de Progress exige une procédure de démarrage ». Où est l'erreur ?

R. Il y a plusieurs possibilités. Cela se produit quand vous avez démarré une session Progress et avez tenté d'accéder à l'éditeur de procédures. Le produit Client/Networking est conçu pour n'exécuter que du code compilé ou du code source crypté. Vous ne pourrez pas compiler de nouveaux programmes avec ce client Progress ; il vous faut le Query/Runtime ou le client L4G. Si vous avez acheté l'un de ces produits et que le message apparaît encore, vérifier les variables **$DLC** et **$PROCFG**, car Progress consulte le fichier **progress.cfg** pour savoir quels produits sont installés, et s'il ne peut trouver ce fichier, il suppose que vous n'avez pas les licences associées. Un fichier **progress.cfg** abimé peut aussi en être à l'origine, ainsi que le paramètre de démarrage de session **–rr**.

© copyright Éditions Eyrolles

9

Installer et gérer PostgreSQL sur Linux

Dans ce chapitre

- Installation de PostgreSQL
- Installation et configuration de la version source de PostgreSQL
- Valider et vérifier l'installation de PostgreSQL
- Gestion et sécurité d'une base de données PostgreSQL
- Ressources et outils sur l'Internet

© copyright Éditions Eyrolles

Introduction

Tous les systèmes informatiques au monde s'appuient sur des bases de données pour ranger et retrouver les informations. Un ordinateur sert à ranger, retrouver et traiter des informations, ce qu'il doit faire très vite, pour faire gagner du temps aux humains. En même temps, le système doit être simple, robuste, rapide, fiable, économique et très facile à utiliser. Une base de données est la partie du système la plus vitale, parce qu'elle stocke les informations critiques de toutes les entreprises. Les industries, telles les télécommunications, celles du secteur de l'automobile, les banques, les compagnie aériennes, etc. ne peuvent fonctionner de manière efficace sans système de base de données. Les systèmes de bases de données les plus répandus reposent sur les spécifications des standards SQL, de l'ISO et de l'ANSI (aux États-Unis). Les spécifications actuelles les plus utilisées dans l'industrie sont les standards SQL ISO/ANSI 1992/98. Sans un standard tel que SQL ANSI/ISO, il serait très difficile pour le client de développer une application d'un côté, et de l'exécuter par ailleurs sur d'autres systèmes de bases de données. L'utilisateur final veut qu'une application soit développée une fois pour toutes, afn de pouvoir passer par SQL ISO, ODBC ou JDBC, et la déployer sur divers systèmes de bases de données.

PostgreSQL est actuellement la plus avancée des bases de données objet relationnelle *open-source* au monde. Postgres, développée à l'origine par le département informatique de l'université de Berkeley, a beaucoup innové dans le domaine des concepts « objet-relationnel ». Il offre, outre le langage SQL92/SQL3, les fonctionnalités suivantes : intégrité des transactions, fonctions définies par l'utilisateur, procédures stockées, déclencheurs et extensibilité de type. PostgreSQL descend en droite ligne du code source originaire de Berkeley ; il fait partie du domaine public, et il est d'accès gratuit. Il existe des millions de bases de données PostgreSQL, installées à titre de serveurs de base de données, tant sur le web que pour des applications. C'est un système de base de données relationnelle objet très sophistiqué (ORDBMS).

PostgreSQL fonctionne sur Solaris, SunOS, HPUX, AIX, Linux, Irix, Digital Unix, BSDi, NetBSD, FreeBSD, SCO UNIX, NEXTSTEP, Unixware et presque toutes les variantes d'UNIX. Un portage sur Microsoft Windows NT a été réalisé à l'aide du progiciel cygwin32 de Cygnus.

Fiche d'identité :

- Titre : Base de données SQL PostgreSQL (Système de gestion de base de données relationnelle objet).
- Version actuelle : 7.0.
- Âge : PostgreSQL a maintenant 15 ans, il a vu le jour en 1985.

© copyright Éditions Eyrolles

- Auteurs : PostgreSQL est développé, depuis 15 ans, par des millions de gens, que ce soit dans les entreprises, les universités ou sur l'Internet.

MySQL est un autre système de base de données *open-source*, presque un progiciel ; il propose des caractéristiques importantes, mais lourdes, telles que les transactions.

PostgreSQL dans le sillage de l'Internet

L'Internet, qui est au départ un projet militaire, est maintenant ouvert au monde entier ; il peut mettre en relation des gens de tous les continents. Il permet d'échanger rapidement des informations, des idées innovantes ou des sujets de recherche depuis tous les points du globe. L'Internet est donc devenu le plus grand « générateur » de logiciel du monde. Les méthodologies de développement de logiciel ont évolué, des sources statiques et propriétaires (*closed-source*) vers le développement *open-source*[1], c'est-à-dire dynamique, évolutif, selon de nouvelles modalités d'échange, et l'Internet facilite un tel système.

L'Internet a permis de créer le système d'exploitation *Linux*, devenu très courant. Devant les progrès de Linux, beaucoup de gens ont opté pour des applications informatiques comme les systèmes de bases de données, les tableurs, etc. Plutôt que de partir de zéro, les développeurs sur l'Internet ont utilisé le code Postgres du domaine public de l'université de Californie et, peu à peu, l'ont enrichi et amélioré : le résultat est PostgreSQL.

L'Internet est un véritable moteur pour PostgreSQL. Les statistiques et les constats d'efficacité d'ordre matériel sont en partie à l'origine du succès de PostgreSQL, car elles favorisent les systèmes *open-source*, tels PostgreSQL ou Linux. Comme la vitesse d'Internet augmente de jour en jour, et que l'Internet est de plus en plus fiable, les systèmes *open-source* sont de plus en plus réactifs. Si l'on soumet le logiciel à des critères fondés sur les principes statistiques, mathématiques et scientifiques, on aboutit au label de « meilleure qualité du logiciel ». Ce sont les systèmes à base de code *open-source* qui obtiennent les meilleurs résultats, car ils bénéficient de l'apport d'Internet, « autoroute de l'information », par définition : PostgreSQL est de ceux-là. Plus il y a de gens à travailler sur le code, plus le logiciel s'améliore. Le modèle de code *open-source* élimine aussi le travail redondant, est donc très économe, y compris du point de vue de la distribution, et il répond aux impé-

1. open-source = ouvert ; l'évolution et l'adaptation de tels logiciels dépend de suggestions, développements et tests, qui sont externes aux premières équipes de développement, et sont acceptés comme un enrichissement. *N.d.T.*

© copyright Éditions Eyrolles

ratifs économiques d'optimisation des ressources nationales et globales. On n'a pas besoin de gaspiller son temps à refaire ce qui est déjà bien fait.

Le modèle de développement *open-source* est à l'origine de nombreuses caractéristiques de PostgreSQL. Le développement du logiciel sur l'Internet est très rapide, car un réseau d'ordinateurs et de gens extrêmement vaste est impliqué dans le processus. Les tendances font apparaître que la plupart des développements de logiciels prennent place sur les autoroutes de l'information, qui parcourent le monde entier. Dans un proche avenir, la croissance d'Internet sera explosive, et cela dépassera de loin la simple adoption de PostgreSQL par l'industrie.

Comme un gros travail a été fait depuis 15 ans sur PostgreSQL, il n'y a aucun intérêt à recréer, de toutes pièces, un système de base de données qui satisfait aux standards SQL ANSI/ISO. Au contraire, il vaut mieux prendre le code existant de PostgreSQL, quitte à lui ajouter certaines fonctionnalités qui lui manquent, ou des améliorations, pour pouvoir s'en servir le plus vite possible.

La demande de produits Internet tels que PostgreSQL devrait croître de manière exponentielle : on est en effet capable de maintenir un haut niveau de qualité et, pour un coût réduit, de fournir, aux utilisateurs et aux développeurs, un environnement très vaste. Les pays qui n'ont guère d'accès aux produits développés sur l'Internet risquent de prendre un sérieux retard et de rater la révolution « cyber-mondiale » liée à la toile. Internet est le plus grand éditeur de logiciel au monde.

Se procurer PostgreSQL

Si l'on envisage d'utiliser PostgreSQL sur un système d'exploitation Linux, on peut acheter les CD-ROM Linux de Red Hat, de Debian ou de Slackware qui contiennent déjà PostgreSQL sous forme de paquetages (code source et binaire), ou bien les acheter en ligne sur l'un de ces sites web :

- Linux System Labs : http://www.lsl.com/
- Red Hat : http://www.redhat.com/
- Cheap Bytes Inc. : http://www.cheapbytes.com/
- Debian : http://www.debian.org/vendors.html

L'organisation PostgreSQL vend aussi un CD-ROM PostgreSQL, qui contient le code source complet et les fichiers binaires destinés à de nombreux systèmes d'exploitation UNIX, ainsi qu'une documentation en ligne : le site principal du CD-ROM PostgreSQL est http://www.postgresql.org

© copyright Éditions Eyrolles

Si l'on veut télécharger depuis l'Internet, visiter les sites suivants :

- RPM source et binaires de PostgreSQL :
  http://www.ramifordistat.net/postgres
- Lamar Owen, pour la maintenance des RPM PostgreSQL :
  lamar.owen@wgcr.org
- RPM source et binaires de PostgreSQL : http://www.postgresql.org
  Cliquer sur *Latest News* et sur *Red Hat RPMs*.
- RPM source et binaires de PostgreSQL :
  http://www.redhat.com/pub/contrib/i386/
  Site ftp : ftp://ftp.redhat.com/pub/contrib/i386/
- Site pour les binaires sur Solaris, HPUX, AIX, IRIX et Linux :
  ftp://ftp.postgresql.org/pub/bindist

Installation rapide de PostgreSQL

Même si PostgreSQL est le plus avancé et le plus sophistiqué des systèmes de base de données au monde, il n'est pas du tout complexe à installer, et de fait, l'installation prend moins de cinq minutes, avec un RPM binaire. PostgreSQL nécessite un minimum de 40 Mo d'espace disque et de 8 Mo de RAM, mais, comme aujourd'hui, même les systèmes PC à bas prix ont souvent un disque dur de 10 Go et 64 Mo RAM, et que le disque dur n'est pas cher... Cette section donne une vue d'ensemble du processus d'installation. Si l'on a une certaine expérience de Linux et des paquetages RPM Red Hat, on pourra effectuer l'installation de PostgreSQL en un rien de temps en se conformant aux étapes qui suivent. Si l'on est novice en Linux/UNIX, se reporter à la section suivante, « Instructions détaillées », qui explique en détail toutes ces étapes.

Sur les systèmes Red Hat Linux, suivre les étapes ci-dessous. Sur Debian Linux, les étapes sont similaires, mais il faut remplacer **.rpm** par **.deb**. Sur Debian Linux, si l'on n'a pas les paquetages PostgreSQL, on peut convertir les paquetages PostgreSQL de Red Hat en paquetages Debian à l'aide du programme **rpm2deb**. Se connecter comme root et passer les commandes **:**

```
# cd /mnt/cdrom/RedHat/RPMS
# man rpm
# ls postgre*.rpm                    pour voir la liste des paquetages
# rpm -qpi postgre*.rpm | less       informations sur les paquetages
# rpm -qpl postgre*.rpm | less       liste des fichiers de chaque paquetage
# ps -auxw | grep -I postmaster
```

© copyright Éditions Eyrolles

Si le postmaster est en fonctionnement, sortir des applications PostgreSQL et arrêter le processus postmaster.

```
# cat /etc/passwd | grep postgres
```

Important. Si l'on constate qu'il existe déjà un utilisateur **postgres**, sauvegarder et nettoyer le répertoire racine de Postgres, **~postgres**, supprimer l'utilisateur **postgres**, ou éventuellement le renommer **postgres2**. Il faut que l'installation soit propre.

Installer les paquetages **postgresql** nécessaires — il faut installer tous les paquetages clients, données et serveur, pour que **pgaccess** fonctionne.

```
# rpm -i postgre*.rpm
```

Pour démarrer PostgreSQL lors du démarrage (*boot*) :

```
# man chkconfig
# chkconfig –add postgresql
```

Pour lancer le serveur Postgres dès maintenant :

```
# /etc/rc.d/init.d/postgresql start
```

Pour donner à **pgaccess** l'accès à l'affichage :

```
# man xhost
# xhost +
# su - postgres
bash$ man createdb
bash$ createdb mydatabase
bash$ man psql
bash$ psql mydatabase
..... in psql press up/down arrow keys for history line editing or give \par
bash$ export DISPLAY=<hostname:>0.0
bash$ man pgaccess
bash$ pgaccess mydatabase
```

Maintenant on peut commencer à taper des commandes SQL dans **psql** ou **pgaccess** !

Pour consulter les FAQ, les guides d'utilisation, de programmation et d'administration, ou les didacticiels, utiliser :

```
bash$ cd /usr/doc/postgresql*
```

Ce répertoire contient tous les documents de PostgreSQL.

© copyright Éditions Eyrolles

Pour lire les fichiers Postscript compressés :

```
bash$ man gv
bash$ gv admin.ps.gz
```

Si les étapes précédentes n'aboutissent pas, chercher d'autres informations sur l'installation à http://www.ramifordistat.net/postgres. Lamar Owen assure la maintenance des RPM PostgreSQL RPM : son adresse e-mail est lamar.owen@wgcr.org. Il faut se familiariser soi-même avec le gestionnaire de paquets RPM de Red Hat pour bien gérer les installations PostgreSQL. Télécharger le manuel *Maximum RPM* de http://www.rpm.org, et consulter le fichier nommé **maximum-rpm.ps.gz** — on peut le lire sur Linux à l'aide de la commande **gv** :

```
bash$ gv maximum-rpm.ps.gz
```

Détails de l'installation

Cette section explique en détail chacune des étapes de la section précédente « Installation rapide de PostgreSQL ».

Insérer le CD-ROM Red Hat Linux le plus récent dans le lecteur, se connecter en tant que root, puis passer les commandes suivantes**:**

```
#  mount /mnt/cdrom
```

qui monte le CD-ROM dans le répertoire **/mnt/cdrom**. Ensuite, aller dans le répertoire qui contient tous les RPM :

```
# cd /mnt/cdrom/RedHat/RPMS
```

Quelques connaissances de base sur les paquetages RPM et la commande Linux **rpm** sont nécessaires : voir la documentation en ligne, qui explique les options et l'usage de **rpm**, à l'aide de la commande **man rpm**.

Les RPM PostgreSQL sont en général nommés **postgresql-aaa-x.x.x-r.rpm**, où **aaa** est le paquetage, comme **jdbc** ou **server, x.x.x** le numéro de version, 7.0.0, par exemple, et **r** la version du paquetage, telle **1**, **2**, **3**, etc. Pour voir la liste des RPM PostgreSQL :

```
# ls postgre*.rpm
postgresql-6.5.2-1.i386.rpm
postgresql-devel-6.5.2-1.i386.rpm
postgresql-jdbc-6.5.2-1.i386.rpm
```

© copyright Éditions Eyrolles

```
postgresql-odbc-6.5.2-1.i386.rpm
postgresql-perl-6.5.2-1.i386.rpm
postgresql-python-6.5.2-1.i386.rpm
postgresql-server-6.5.2-1.i386.rpm
postgresql-tcl-6.5.2-1.i386.rpm
postgresql-test-6.5.2-1.i386.rpm
```

Pour consulter les informations sur les RPM PostgreSQL, utiliser l'option **-i rpm** et la commande **pipe it to less**, pour les voir page par page. Voir la page du manuel en ligne en tapant **man less**.

```
# rpm -qpi postgre*.rpm | less
Name        : postgresql                    Relocations : /usr
Version     : 6.5.2                         Vendor      : Red Hat Software
Release     : 1                             Build Date  : Sun 26 Sep 1999
                                                          03:41:44 PM CDT
Install date: (not installed)               Build Host  : porky.devel.redhat.com
Group       : Applications/Databases        Source RPM  : postgresql-6.5.2-1.src.rpm
Size        : 7354692                       License     : BSD
Packager    : Red Hat Software <http://developer.redhat.com/bugzilla>
URL         : http://www.ramifordistat.net/
Summary     : The PostgreSQL server programs.
Description : Postgresql includes the programs needed to create and run
a PostgreSQLserver, which will in turn allow you to create and maintain
PostgreSQL databases.  PostgreSQL is an advanced Object-Relational database
management system (DBMS) that supports almost all SQL constructs.
```

Quelquefois, il vaut mieux vérifier la liste des fichiers contenus dans les paquetages RPM. Pour voir la liste des fichiers du RPM PostgreSQL RPM :

```
# rpm -qpl postgre*.rpm | less
```

L'option **-q** signifie mode de requête, **-p** paquetage et **-l** liste des fichiers.

Si PostgreSQL existe déjà sur le système et que la base de données fonctionne sur le port par défaut, il faut d'abord arrêter le processus postmaster. Pour voir s'il existe un processus PostgreSQL, taper :

```
# ps -auxw | grep -I postmaster
```

Si le postmaster est actif, sortir de toutes les applications PostgreSQL et arrêter le processus postmaster. Pour ce faire, utiliser la commande :

```
# /etc/rc.d/init.d/postgresql stop
```

L'installation de PostgreSQL doit toujours être un « clean slate » c'est-à-dire qu'il faut débarrasser le système de toute installation antérieure. Si l'on a déjà installé PostgreSQL, il faut la transférer ou la renommer de sorte qu'il n'y ait

© copyright Éditions Eyrolles

aucun « conflit de nom ». En général, PostgreSQL est installé sous l'id d'utilisateur UNIX **postgres**. Pour vérifier s'il y a déjà un utilisateur UNIX nommé **postgres** sur le système Linux, taper :

```
# cat /etc/passwd | grep postgres
postgres:x:40:233:PostgreSQL Server:/var/lib/pgsql:/bin/bash
```

S'il y a déjà un utilisateur **postgres**, sauvegarder et nettoyer le répertoire racine de Postgres, nommé **~postgres**, et supprimer l'utilisateur UNIX **postgres**, ou le renommer, par exemple, **postgres2**.

Installer les paquetages **postgresql** nécessaires ; si l'on a assez d'espace disque, installer tous les RPM PostgreSQL.

```
# rpm -i postgre*.rpm
```

Si l'on veut que la base de données PostgreSQL démarre automatiquement au démarrage du système Linux, ajouter PostgreSQL aux scripts de démarrage du système à l'aide de la commande Linux **chkconfig**. Pour lire le manuel en ligne, utiliser **man chkconfig** ; la commande

```
# chkconfig –add postgresql
```

ajoutera les liens suivants au fichier **/etc/rc.d/init.d/postgresql** du répertoire **/etc/rc?.d** :

```
init.d/postgresql
rc0.d/K15postgresql
rc1.d/K15postgresql
rc2.d/K15postgresql
rc3.d/S85postgresql
rc4.d/S85postgresql
rc5.d/S85postgresql
rc6.d/K15postgresql
```

À partir de maintenant, tous les fichiers de système PostgreSQL sont en place, on démarre le serveur Postgres avec la commande

```
# /etc/rc.d/init.d/postgresql start
```

qui lance le processus Linux **postmaster**, lequel va être à l'écoute des requêtes sur le port TCP **5432**. Pour consulter le processus, taper

```
# ps -auxw | grep -I postmaster
```

Maintenant, on peut utiliser les applications interfaces de PostgreSQL. Les deux interfaces utilisateur de PostgreSQL les plus souvent utilisées sont

© copyright Éditions Eyrolles

pgaccess et **psql**. **psql** est un terminal interactif en ligne de commande, et **pgaccess** une application graphique X Window écrite en Tcl/Tk, extrêmement convivial : on peut utiliser la souris pour cliquer sur des fonctions diverses, telles que créer des tables et des bases de données, exécuter des requêtes et des rapports, etc. Comme **pgaccess** est une application X Window, on doit donner aux autres utilisateurs UNIX la permission d'utiliser l'affichage du serveur X Window.

Pour ce faire, taper

```
# xhost +
```

Le + après **xhost** signifie que l'on donne accès à l'affichage à tous les utilisateurs UNIX du système. Pour plus d'informations, se reporter à un manuel standard de X Window, ou consulter la page du manuel en ligne :

```
# man xhost
```

Se connecter en tant que **postgres** à l'aide de la commande :

```
# su - postgres
```

Voir **man su** pour les détails de la commande **su** (setuser).

Maintenant, l'invite # disparaît au profit de **bash$**. Taper :

```
bash$ id
postgres(uid=10 gid=4)
```

On est prêt à créer une première base de données ! Cela se fait avec la commande **createdb**. Voir la page de manuel en ligne avec **man createdb.**

```
bash$ createdb mydatabase
```

crée une base de données nommée **mydatabase**. Quand celle-ci est créée, on peut y accéder avec l'interface **psql.**

Pour consulter la documentation en ligne, taper :

```
bash$ man psql
```

Pour accéder à **mydatabase**, entrer la commande :

```
bash$ psql mydatabase
```

qui lance le programme **psql**, avec l'invite **psql.** Dans **psql**, on utilise les flèches pour éditer une ligne de l'historique, ou bien la commande **\s.** Mainte-

© copyright Éditions Eyrolles

nant, on peut entrer toutes les commandes SQL standard à l'invite **psql.** Se reporter à un guide ou à un manuel SQL pour plus d'informations. On sort du programme **psql** par la commande **\q**, qui ramène à l'invite **bash$.**

Pour utiliser l'interface **pgaccess** de PostgreSQL, il faut exporter une variable d'environnement Linux appelée **DISPLAY** :

```
bash$ export DISPLAY=<hostname>:0.0
```

où **hostname** est le nom de l'hôte Linux donné dans le fichier **/etc/hosts** ou l'adresse IP de l'hôte. Par exemple, on peut taper :

```
bash$ export DISPLAY=elmo:0.0
```

ou bien :

```
bash$ export DISPLAY=198.210.32.110:0.0
```

où **198.210.32.110** est l'adresse IP de l'hôte nommé **elmo**. Pour consulter le manuel de **pgaccess**, taper :

```
bash$ man pgaccess
```

Pour accéder à la base de données **mydatabase**, taper :

```
bash$ pgaccess mydatabase
```

qui lancera l'application graphique adéquate sur le système X Window (KDE ou GNOME). Il y a aussi un bouton d'aide, qui donne plus de détails sur **pgaccess**.

Maintenant, on peut passer des commandes SQL avec **psql** ou **pgaccess** !

Sur un système Red Hat Linux, toute la documentation est installée dans le répertoire **/usr/doc**. Pour consulter les FAQ, les guides d'utilisation, de programmation et d'administration, et les didacticiels, taper :

```
bash$ cd /usr/doc/postgresql*
```

On peut lire un fichier Postscript compressé à l'aide de la commande :

```
bash$ gv admin.ps.gz
```

Consulter la documentation en ligne de la commande **gv :**

```
bash$ man gv
```

© copyright Éditions Eyrolles

Si les étapes précédentes n'aboutissent pas, on peut obtenir d'autres informations à http://www.ramifordistat.net/postgres. Lamar Owen assure la maintenance des RPM PostgreSQL : son adresse e-mail est lamar.owen@wgcr.org. Il faut se familiariser soi-même avec le gestionnaire de paquet RPM de Red Hat pour bien gérer les installations PostgreSQL. Télécharger le manuel *Maximum RPM* de http://www.rpm.org, et consulter le fichier nommé **maximum-rpm.ps.gz** — on peut le lire sur Linux à l'aide de la commande **gv maximum-rpm.ps.gz**.

Installation de la distribution source

Ensuite, vous aurez besoin des outils GNU **gcc** et **make**. On peut vérifier si l'on dispose de ces outils à l'aide des commandes suivantes :

```
shell$ which gcc
/usr/bin/gcc
shell$ which make
/usr/bin/make
```

Le **gcc** installé doit être en version 2.81 ou supérieure. Vérifier la version par la commande :

```
shell$ gcc –v
Reading specs from /usr/lib/gcc-lib/i386-redhat-linux/egcs-2.91.66/specs
gcc version egcs-2.91.66 19990314/Linux (egcs-1.1.2 release)
```

Si ce n'est pas la bonne version, se reporter à la documentation de la distribution Linux pour la mettre à jour. Consulter le site GNU (http://www.gnu.org), si l'on a besoin d'aide pour installer les outils. Les commandes ont été testées avec une version 5.2 de Red Hat Linux, utilisant le shell **Bash.** Sauf indication contraire, elles fonctionneront probablement sur la plupart des systèmes. Les options des commandes telles que **ps** et **tar** varient sensiblement selon les plates-formes. Le mieux est de se fier au bon sens avant d'employer ces commandes.

Si l'on n'a pas la distribution PostgreSQL, la prendre sur ftp.postgresql.org, puis la décompresser comme ceci :

```
$ gunzip postgresql-7.0.tar.gz
$ tar -xvf postgresql-7.0.tar
$ mv postgresql-7.0 /usr/src
```

© copyright Éditions Eyrolles

Avant de démarrer

La construction de PostgreSQL nécessite la commande GNU **make**, qui ne fonctionne pas avec d'autres programmes **make**. Sur les systèmes Linux GNU, **make** de GNU est l'outil par défaut ; sur d'autres systèmes, il se peut qu'il soit installé sous le nom de **gmake**. Par la suite, nous utiliserons **gmake** pour nous référer à **make** de GNU, indépendamment du nom qu'il porte sur le système. Pour vérifier la présence de **make** GNU, entrer :

```
$ gmake –version
```

Pour récupérer **make** de GNU, voir ftp://ftp.gnu.org. On trouve des informations à jour à propos des plates-formes à http://www.postgresql.org/docs/admin/ports.htm. En général, les plates-formes compatibles UNIX avec les bibliothèques actuellement utilisées sont capables d'exécuter PostgreSQL. Dans le sous-répertoire **doc** de la distribution figurent plusieurs documents README et FAQ spécifiques de la plate-forme, que l'on peut consulter si l'on rencontre un problème.

Bien que l'on puisse exécuter PostgreSQL avec la mémoire minimale exigée de 8 Mo, la vitesse s'améliore nettement quand la mémoire est étendue à 96 Mo, ou plus. La règle est que l'on n'a jamais trop de mémoire.

Vérifier aussi que l'espace disque est suffisant : il faut environ 30 Mo pour l'arborescence du source pendant la compilation, et 5 Mo pour le répertoire d'installation. Une base de données, vide, occupe environ 1 Mo ; sinon, il faut prévoir cinq fois plus que l'espace pris par un fichier texte pour le même volume de données. Si l'on exécute les tests de non-régression, prévoir un espace temporaire d'environ 20 Mo. Pour vérifier l'espace disque, utiliser la commande

```
$ df -k
```

Étant donné le prix actuel des disques durs, il faudra probablement étudier une configuration assez grande et rapide avant de mettre une base de données en production.

Procéder étape par étape

Lors d'une nouvelle installation, ou lors de la mise à jour d'une version précédente de PostgreSQL, procéder comme suit :

1. Créer le compte du super-utilisateur PostgreSQL, qui sera celui du serveur. Pour la production, créer un compte séparé, sans privilèges (on utilise couramment **postgres**). Si l'on souhaite seulement se servir du

© copyright Éditions Eyrolles

compte, sans avoir d'accès root, un compte utilisateur suffit.

Exécuter PostgreSQL en tant que root, bin ou tout autre compte doté de droits d'accès spéciaux présente un risque pour la sécurité : ce n'est donc pas recommandé. On n'a pas besoin de faire l'installation sous ces comptes particuliers, bien que ce soit possible. Lorsque les droits du super-utilisateur de base de données sont nécessaires, c'est indiqué dans la procédure.

2. S'il ne s'agit pas d'une mise à jour, passer à l'étape 4.

   Il faut maintenant sauvegarder la base de données existante — il s'agit de l'installation de la base de données la plus récente, au-delà de la version 6.0 — taper

```
$ pg_dumpall > db.out
```

   Pour conserver les identifiants d'objets (OID), utiliser l'option –**o** quand on exécute **pg_dumpall**. Cependant, sauf raison particulière (par exemple, si l'on se sert des OID comme clés dans les tables), il ne faut pas le faire.

   Bien utiliser la commande **pg_dumpall** de la version en cours ; ne pas utiliser le script **pg_dumpall** de la version 6.0, sinon tout deviendra la propriété du super-utilisateur PostgreSQL. Dans ce cas, il faudra récupérer un **pg_dumpall** d'une version 6.x.x plus récente, sachant que le **pg_dumpall** de la version 7.0 ne fonctionne pas sur les bases de données d'une version antérieure. Si l'on migre depuis une version antérieure à la version 1.09 de Postgres95, il faut sauvegarder la base de données, installer Postgres95 v1.09, restaurer la base et la sauvegarder à nouveau.

Attention. Il faut s'assurer que la base de données n'est pas en cours de mise à jour lors de la sauvegarde. Si nécessaire, arrêter postmaster, éditer les permissions contenues dans le fichier **/usr/local/pgsql/data/pg_hba.conf** pour n'autoriser qu'un seul utilisateur, puis relancer postmaster.

3. Si l'on met à jour un système existant, arrêter le serveur de base de données. Entrer la commande

```
$ ps ax | grep postmaster
```

   Pour obtenir la liste des numéros de processus en cours :

```
1032  ?      S  0:00  /usr/bin/postmaster -i -S -D/var/lib/pgseql
1089 pts/3 S  0:00  grep postmaster
```

   Taper la ligne suivante en remplaçant **pid** par l'id du processus postmaster (**1032** dans ce cas — ne pas utiliser l'ID du processus **grep postmaster**).

```
$ kill pid
```

© copyright Éditions Eyrolles

Remarque. Sur les systèmes où PostgreSQL démarre lors du boot, il y a probablement un fichier de démarrage qui fait la même chose. Par exemple, sur un système Red Hat Linux, on trouvera que la commande :

```
$ /etc/rc.d/init.d/postgres.init stop
```

fonctionne. Il faut aussi mettre de côté les anciens répertoires. Entrer

```
$ mv /usr/local/pgsql /usr/local/pgsql.old
```

ou bien les chemins modifiés.

4. Configurer le code source pour le système. C'est à cette étape que l'on peut spécifier le chemin d'installation réel pour le processus de construction et choisir ce qui doit être installé. Se placer dans le sous-répertoire **src** et entrer

```
$. /configure
```

suivi de toutes les options. Pour une première installation, on peut ne rien mettre. Pour consulter la liste complète des options, taper :

```
. /configure –help
```

Voici quelques-unes des options les plus couramment utilisées :

- **prefix=BASEDIR** : choisit un répertoire de base à part pour installer PostgreSQL. Par défaut, c'est **/usr/local/pgsql**.
- **enable-locale** : permet d'utiliser les variables locales.
- **enable-multibyte** : autorise l'utilisation de l'encodage multi-caractères. Cela concerne surtout le japonais, le coréen ou le chinois.
- **with-perl** : construit l'interface Perl. Noter que l'interface Perl est installée à la place habituelle des modules Perl (en général, sous **/usr/lib/perl**), mais il faut un accès root pour utiliser cette option.
- **with-odbc :** construit le pilote ODBC.
- **with-tcl** : construit les bibliothèques d'interface et les programmes qui ont besoin de Tcl/Tk, dont libpgtcl, pgtclsh et pgtksh.

5. Compiler le programme ; taper

```
$ gmake
```

Le processus de compilation peut durer assez longtemps : de 10 minutes à une heure, selon les cas. La dernière ligne affichée sera, espérons-le :

```
All of PostgreSQL is successfully made. Ready to install.
```

Se souvenir que **gmake** peut s'appeler **make**, selon le système.

© copyright Éditions Eyrolles

6. Installer le programme. Taper

```
$ gmake install
```

7. Indiquer au système comment trouver les nouvelles bibliothèques partagées : la méthode varie selon les plates-formes. Celle qui fonctionne partout consiste à créer la variable d'environnement **LD_LIBRARY_PATH** :

```
$ LD_LIBRARY_PATH=/usr/local/pgsql/lib
$ export LD_LIBRARY_PATH
```

On peut mettre ces commandes dans un fichier de démarrage du shell, tel que **/.bash_profile**.

Sur certains systèmes, la meilleure méthode est celle qui suit, mais on doit avoir un accès root. Éditer le fichier **/etc/ld.so.conf** pour lui ajouter la ligne

```
/usr/local/pgsql/lib
```

et exécuter la commande **/sbin/ldconfig**. En cas de doute, se reporter au manuel de référence du système. Si l'on obtient le message

```
./psql : error in loading shared libraries
libpq.so.2.1 : cannot open shared object file : No such file or directory
```

c'est que l'étape précédente était nécessaire : il suffit alors de l'exécuter.

8. Faire l'installation de la base de données. Pour cela, se connecter sous le compte du super-utilisateur PostgreSQL — avec root, cela ne fonctionne pas.

```
$ mkdir /usr/local/pgsql/data
$ chown postgres /usr/local/pgsql/data
$ su - postgres
$ /usr/local/pgsql/initdb -D /usr/local/pgsql/data
```

L'option –**D** spécifie l'emplacement où les données seront rangées. On peut utiliser un chemin quelconque — il n'a pas besoin d'être dans le répertoire d'installation. S'assurer quand même que le compte du super-utilisateur peut écrire dans ce répertoire (ou le créer) avant de démarrer **initdb**. Si l'on a fait l'installation, jusqu'à maintenant en tant que super-utilisateur PostgreSQL, il faut se connecter temporairement en tant que root pour créer le répertoire des données.

9. L'étape précédente demande de démarrer le serveur de base de données. La commande

```
$ /usr/local/pgsql/bin/postmaster -D /usr/local/pgsql/data
```

© copyright Éditions Eyrolles

démarre le serveur en tâche de premier plan. Pour le démarrer en tâche de fond, utiliser l'option –**S.**

10. Si l'on migre une installation existante, restaurer les données à l'aide de :

```
$ /usr/local/pgsql/bin/psql < db.out
```

On peut aussi récupérer les anciens fichiers pg_hba.conf et tous les autres fichiers utilisés pour l'authentification, tels les fichiers de mots de passe.

Ceci termine l'installation au sens strict. Pour vous rendre la vie plus facile et plus agréable, n'hésitez pas à essayer les suggestions ci-après.

- On se facilite la tâche en initialisant quelques variables d'environnement. D'abord, pour inclure **/usr/local/pgsql/bin** (ou son équivalent) dans son PATH, ajouter la ligne qui suit dans le fichier de démarrage du shell, par exemple **/.bash_profile** (ou dans **/etc/profile**, si on veut l'affecter à chaque utilisateur) :

```
PATH=$PATH:/usr/local/pgsql/bin
```

- De plus, si l'on met PGDATA dans l'environnement du super-utilisateur PostgreSQL, on peut omettre l'option –**D** dans les commandes **postmaster** et **initdb**.
- Pour installer le manuel **man** et la documentation HTML, entrer :

```
$ cd /usr/src/pgsql/postgresql-7.0/doc
$ gmake install
```

- Les fichiers seront installés dans les répertoires **/usr/local/pgsql/doc** et **/usr/local/pgsql/man**. Pour donner au système l'accès à la documentation man, il faut ajouter la ligne suivante à un fichier de démarrage du shell :

```
MANPATH=$MANPATH:/usr/local/pgsql/man
```

- La documentation est disponible aussi en format Postscript. Si l'on a une imprimante Postscript, ou si la machine est déjà initialisée pour imprimer les fichiers Postscript à l'aide d'un filtre, on imprime le guide de l'utilisateur comme ceci :

```
$ cd /usr/local/pgsql/doc
$ gunzip -c user.ps.tz | lpr
```

- Voici comment faire dans le cas où **ghostscript** est installé sur le système et que l'on utilise une imprimante laser :

```
$ alias gshp='gs -sDEVICE=laserjet -r300 -dNOPAUSE'
$ export GS_LIB=/usr/share/ghostscript:/usr/share/ghostscript/fonts
```

© copyright Éditions Eyrolles

```
$ gunzip user.ps.gz
$ gshp -sOUTPUTFILE=user.hp user.ps
$ gzip user.ps
$ lpr -l -s -r manpage.hp
```

- En cas de doute, consulter les manuels ou un expert. Le guide de l'administrateur sera probablement à lire en premier si l'on est novice sur PostgreSQL : il contient toutes les informations sur la manière de créer les utilisateurs des bases de données et de mettre en œuvre l'authentification.
- En général, on souhaite configurer l'ordinateur de sorte qu'il démarre le serveur de base de données automatiquement, quand il est mis sous tension ; mais ce n'est pas nécessaire, car le serveur PostgreSQL peut être lancé depuis un compte non privilégié, sans intervention du compte root.
- Les systèmes ont des conventions différentes de démarrage des démons au moment du boot, aussi on sera bien avisé de s'en informer. La plupart des systèmes ont un fichier appelé **/etc/rc.local** ou **/etc/rc.d/rc.local** : c'est certainement l'un des meilleurs endroits pour une telle commande. Quoi que l'on fasse, **postmaster** doit être exécuté par le super-utilisateur PostgreSQL (postgres) et non par root ou tout autre utilisateur. Par conséquent, il faudra toujours formuler les lignes de commande selon la syntaxe **su -c '...' postgres**.
- Conserver une trace de la sortie du serveur. Pour cela, au démarrage du serveur, essayer :

```
nohup su -c 'postmaster -D /usr/local/pgsql/data > server.log 2>&1' postgres &
```

Voici d'autres suggestions, à appliquer selon les systèmes d'exploitation.

- Editer le fichier **rc.local** sur NetBSD, ou **rc2.d** sur SPARC Solaris 2.5.1 pour y ajouter la ligne suivante :

```
su postgres -c "/usr/local/pgsql/bin/postmaster -S -D /usr/local/pgsql/data"
```

- Dans FreeBSD 2.2-RELEASE, éditer **/usr/local/etc/rc.d/pgsql.sh** pour y ajouter les lignes suivantes, puis modifier ses attributs par les commandes **chmod 755** et **chown root:bin**.

```
# !/bin/sh
[ -x /usr/local/pgsql/bin/postmaster ] && {
   su -l pgsql -c 'exec /usr/local/pgsql/bin/postmaster
      -D/usr/local/pgsql/data
      -S -o -F > /usr/local/pgsql/errlog' &
   echo -n ' pgsql'
}
```

© copyright Éditions Eyrolles

On peut placer les sauts de ligne comme on le voit ici. Le shell est assez intelligent pour passer outre une fin de ligne s'il rencontre une expression inachevée. L'exec ne tient compte que d'un seul niveau du shell sous le processus postmaster, de sorte que le parent est **init**.

- Dans Red Hat Linux, ajouter un fichier **/etc/rc.d/init.d/postgres.init**, qui se comporte comme l'exemple de **contrib/linux/**. Créer un lien symbolique vers ce fichier dans **/etc/rc.d/rc5.d/S98postgres.init**.
- Exécuter les tests de non-régression pour vérifier que PostgreSQL fonctionne sur la machine comme prévu. C'est impératif avant de mettre un serveur en production. Le fichier **/usr/src/pgsql/postgresql-7.0/src/test/regress/README** contient toutes les instructions nécessaires pour exécuter et interpréter les tests de non-régression.

Valider l'installation de PostgreSQL

Les RPM d'exemples contiennent des programmes qui permettent de valider l'installation de PostgreSQL sur un site et de tester diverses interfaces vers PostgreSQL.

Les exemples de PostgreSQL

Le répertoire d'exemples **postgresql** s'installe depuis l'une de ces sources :

- CD-ROM Linux : postgresql-*examples.rpm ;
- Télécharger postgresql-*examples.rpm depuis http://www.aldev.8m.com ou http://www.aldev.webjump.com ;
- Dans l'arborescence du code source de PostgreSQL, postgresql*.src.rpm : chercher les répertoires contenant des exemples, des résultats de test ou des guides d'utilisation.

Tester l'interface Tcl/Tk

Le programme **pgaccess** est un exemple d'interface Tcl/Tk. Il fonctionne sous X Window, avec KDE, GNOME ou une fenêtre Motif. Consulter le fichier **/usr/bin/pgaccess** à l'aide d'un éditeur tel que **vi**, **gvim** ou **view** :

```
# man xhost
# xhost +
# su - postgres
bash$ view /usr/bin/pgaccess
bash$ export DISPLAY=<hostname of your machine:>0.0
bash$ createdb mydb
bash$ pgaccess mydb
```

© copyright Éditions Eyrolles

Une belle interface graphique présente différents boutons pour créer des formulaires, des tables, des requêtes, des rapports, etc. Pour plus d'informations sur **pgaccess**, visiter le site http://www.flex.ro/pgaccess. Si vous avez des commentaires ou des suggestions à faire, n'hésitez pas à envoyer un e-mail à teo@flex.ro.

Tester l'interface Python — PyGreSQL

Après avoir installé les exemples de PostgreSQL, procéder comme suit :

```
bash$ cd /usr/lib/pgsql/python
bash$ createdb thilo
bash$ psql thilo
thilo=> create table test (aa char(30), bb char(30) );
bash$ /usr/bin/python
>>> import _pg
>>> db = _pg.connect('thilo', 'localhost')
>>> db.query("INSERT INTO test VALUES ('ping', 'pong')")
>>> db.query("SELECT * FROM test")
eins|zwei
----+----
ping|pong
(1 row)
>>>CTRL+D
bash$
```

Cela semble fonctionner, maintenant installons-le proprement :

```
bash$ su - root
# cp /usr/lib/pgsql/python/_pg.so /usr/lib/python1.5/lib-dynload
```

Tester l'interface Perl

Après avoir installé les exemples de PostgreSQL, entrer les commandes :

```
bash$ cd /usr/doc/postgresql-7.0.0/examples/perl5
bash$ perl. /example.pl
```

Important. Si la commande ne fonctionne pas, essayer la suivante. La variable globale @INC doit inclure le module Pg.pm du répertoire **site_perl**, aussi on utilise l'option **-i** :

```
bash$ perl -I/usr/lib/perl5/site_perl/5.005/i386-linux-thread. /example.pl
```

Perl est lancé et accède à la base de données PostgreSQL ! Pour plus d'informations sur l'utilisation de l'interface Perl, consulter le fichier **example.pl.**

© copyright Éditions Eyrolles

Tester les interfaces libpq++ et libpq

Après avoir installé les exemples de PostgreSQL, entrer les commandes :

```
bash$ cd /usr/doc/postgresql-7.0.0/examples/libpq++
bash$ su root   --> pour changer la propriété des exemples
# chown -R postgres /usr/doc/postgresql-7.0.0/examples
# exit
bash$ g++ testlibpq0.cc -I/usr/include/pgsql -I/usr/include/pgsql/libpq++
-lpq++ -lpq -lcrypt
bash$. /a.out
```

Ignorer les messages d'erreur s'il y en a, comme ci-dessous :

```
> create table foo (aa int, bb char(60));
No tuples returned...
status = 1
Error returned: fe_setauthsvc: invalid name: , ignoring...
> insert into foo values ('4535', 'vasu');
No tuples returned...
status = 1
Error returned: fe_setauthsvc: invalid name: , ignoring...
> select * from foo;
aa    |bb    |
------|------|
4535 |vasu |
Query returned 1 row.
>CTRL+D
bash$
```

Dans cet exemple, on a exécuté directement les interfaces C/C++ de PostgreSQL.

Tester les interfaces Java

Après avoir installé les exemples PostgreSQL, installer ensuite :

- le JDK, jdk-*glibc*.rpm : voir ftp://ftp.redhat.com/pub/contrib/i386 ou http://www.blackdown.org
- postgresql-jdbc-*.rpm : voir ftp://ftp.redhat.com/pub/contrib/i386

puis passer les commandes

```
bash$ cd /usr/doc/postgresql-7.0.0/examples/jdbc
bash$ echo $CLASSPATH
```

qui doivent afficher

```
CLASSPATH=/usr/lib/jdk-x.x.x/lib/classes.zip
```

© copyright Éditions Eyrolles

où x.x.x est le numéro de version correct :

```
bash$ export CLASSPATH=$CLASSPATH::./usr/lib/pgsql/jdbc7.0-1.2.jar
```

Editer les fichiers psql.java et mettre en commentaire la ligne du paquetage :

```
bash$ javac psql.java
bash$ java psql jdbc:postgresql:template1 postgres < password>[1]
select * from pg_tables;

tablename               tableowner   hasindexes   hasrules
pg_type postgres        true         false        false
pg_attribute postgres   true         false        false
[2]
CTRL+C
bash$
```

On a exécuté directement l'interface Java de PostgreSQL.

Tester l'interface ecpg

Après avoir installé les exemples de PostgreSQL, entrer les commandes.

```
bash$ cd /usr/doc/postgresql-7.0.0/examples/ecpg
bash$ ecpg test1.pgc -I/usr/include/pgsql
bash$ cc test1.c -I/usr/include/pgsql -lecpg -lpq -lcrypt
bash$ createdb mm
bash$. /a.out
```

On a exécuté Embedded C-SQL dans la base de données PostgreSQL.

Tester l'interface ODBC

Chercher le pilote ODBC win32 pgsql sur http://www.insightdist.com/psqlodbc/. Voir aussi dans /usr/lib/libpsqlodbc.a.

Tester le pilote à l'aide d'un programme Windows qui accède à la base de données PostgreSQL.

Tester l'interface MPSQL Motif-Worksheet

Chercher le RPM sur http://www.mutinybaysoftware.com et suivre les instructions.

Tester les exemples SQL, les types définis par l'utilisateur et les fonctions

Après avoir installé les exemples de PostgreSQL, entrer les commandes :

© copyright Éditions Eyrolles

```
bash$ cd /usr/doc/postgresql-7.0.0/examples/sql
bash$ make
bash$ sql_exe
```

Vérifier l'installation de PostgreSQL

Pour vérifier la qualité de PostgreSQL, exécuter le paquetage de tests de non-régression. Se connecter en tant que root :

```
#  su - root
# rpm -i postgresql*test.rpm
```

Consulter aussi le fichier README ou installer l'arborescence du code source qui contient le répertoire **regress**

```
# rpm -i postgresql*.src.rpm
# cd /usr/src/redhat/SPECS
# more postgresql*.spec
```

pour savoir quel paquetage RPM système on doit installer. Exécuter ce qui suit pour préparer le paquetage :

```
# rpm -bp postgresql*.spec
```

Le test de non-régression a besoin des fichiers make et de certains fichiers d'en-tête tels que ***fmgr*.h**, construits à l'aide de la commande :

```
# rpm –short-circuit -bc postgresql*.spec
```

Interrompre la commande par Ctrl+C, sitôt qu'apparaît **make -C common SUBSYS.o** : la configuration est alors terminée et les fichiers make et en-têtes sont créés. On n'a pas besoin d'aller plus loin :

```
# cd /usr/src/redhat/BUILD
# chown -R postgres postgresql*
# su - postgres
bash$ cd /usr/src/redhat/BUILD/postgresql-7.0.0/src/test/regress
bash$ more README
bash$ make clean; make all runtest
bash$ more regress.out
```

© copyright Éditions Eyrolles

Appliquer les corrections d'urgence

Certaines corrections de bogues critiques ont été apportées après que la version GA de PostgreSQL eut été distribuée. Ces correctifs optionnels s'appliquent selon les besoins de l'application. D'abord, se placer dans le répertoire source de **postgresql.**

```
# su - root
# rpm -i postgresql*.src.rpm
# cd /usr/src/postgresql7.0.0
# man patch
# patch -p0 < patchfile
# make clean
# make
```

Les fichiers correctifs se trouvent dans les correctifs de PostgreSQL sur le site ftp://ftp.postgresql.org/pub/patches.

Comment s'assurer de la fiabilité de PostgreSQL ?

Une manière quasi-scientifique de valider la fiabilité du système PostgreSQL : la distribution propose un paquetage de test de non-régression (**src/test/regress**), qui vérifie les opérations SQL standard ainsi que les capacités d'extension de PostgreSQL. Les paquetages de test contiennent déjà des centaines de programmes de test en SQL.

On utilise la puissance d'ordinateurs ultra-rapides pour valider PostgreSQL, à la place d'un cerveau humain. Les ordinateurs peuvent exécuter des tests de non-régression du logiciel des millions, voire des milliards de fois plus vite qu'un humain, donc exécuter des milliards de tests SQL en un temps record. Dans un avenir proche, la vitesse des ordinateurs sera des millions et des millions de fois plus rapide que le cerveau humain ! Il est donc tout à fait pertinent d'utiliser la puissance des ordinateurs pour valider un logiciel.

On peut au besoin enrichir les tests, et les déposer sur le site web principal de PostgreSQL si l'on estime qu'ils peuvent être utiles à d'autres personnes. Les paquetages de tests de non-régression renforcent la confiance des clients et facilitent le déploiement de PostgreSQL sur les systèmes de production.

Le paquetage de tests de non-régression est considéré comme une base technique très solide, qui fait l'unanimité entre les développeurs et les utilisateurs finaux. Les développeurs de PostgreSQL utilisent beaucoup le paquetage de tests de non-régression pendant la période de développement, et juste avant de livrer le logiciel au public, en général.

© copyright Éditions Eyrolles

Les possibilités de PostgreSQL sont directement reflétées par le paquetage de tests de non-régression. Si une fonctionnalité, une partie de la syntaxe ou une caractéristique est présente dans le paquetage de tests de non-régression, cela veut dire qu'elles sont valides ; sinon, c'est qu'elles ne sont pas supportées par PostgreSQL ! On peut les vérifier et les ajouter au paquetage de tests de non-régression.

Répertoires du système

Si vous avez installé le RPM PostgreSQL, le répertoire **pgsql** sera dans **/var/lib/pgsql**, et le répertoire **doc** dans **/usr/doc/postgresql-x-x-x**, où **x-x-x** est le numéro de version de PostgreSQL. Par conséquent, il faut ajouter ce répertoire au chemin de commande du shell. Si l'on utilise une variante du shell C de Berkeley, comme **csh** ou **tcsh**, on ajoutera

```
set path = ( /usr/local/pgsql/bin path )
```

dans le fichier de login du répertoire racine (*home*). Si l'on utilise une variante du shell Bourne, comme **sh**, **ksh** ou **Bash**, on ajoutera

```
PATH=/usr/local/pgsql/bin:$PATH
export PATH
```

au fichier **.profile** du répertoire racine.

À partir d'ici, nous supposons que l'on a ajouté le répertoire **bin** de Postgres au chemin. De plus, nous ferons fréquemment référence à la « mise à jour d'une variable de shell » ou à la « mise à jour d'une variable d'environnement » dans la suite de ce chapitre. Si l'on n'a pas bien compris le dernier paragraphe sur la modification du chemin de recherche, consulter les pages du manuel UNIX qui décrivent le shell, avant d'aller plus loin.

Si l'on n'a rien initialisé — dans la méthode par défaut — il reste quelques points à régler. Par exemple, si la machine serveur de base de données est une machine distante, il faut initialiser la variable d'environnement **PGHOST** avec le nom de cette machine ; la variable d'environnement **PGPORT** doit aussi être renseignée. Le fil directeur est : si l'on essaie de démarrer un programme d'application et qu'il ne peut pas se connecter au postmaster, il faut revenir en arrière et s'assurer qu'il est initialisé correctement dans l'environnement.

© copyright Éditions Eyrolles

Utiliser Postgres sur UNIX/Linux en environnement runtime

Toutes les commandes Postgres qui sont exécutées directement depuis le shell UNIX se trouvent dans le répertoire ...**/bin**. Pour faciliter l'exécution de ces commandes, mettre ce répertoire dans le chemin de recherche.

Il existe, sur chaque site, une collection de catalogues système qui comprennent une classe (**pg_user**) contenant une instance par utilisateur Postgres valide. L'instance spécifie un ensemble de privilèges Postgres, tels que la capacité d'agir en tant que super-utilisateur Postgres, celle de créer ou de détruire des bases de données ou celle de mettre à jour les catalogues du système. Un utilisateur UNIX ne peut rien faire avec Postgres tant qu'une instance appropriée n'est pas installée dans cette classe. On obtient d'autres informations sur les catalogues de système en exécutant des requêtes sur les classes appropriées.

Démarrer postmaster

On ne peut rien faire d'une base de données si le processus postmaster n'est pas lancé. L'administrateur du site devra garder à l'esprit nombre de tâches et de vérifications à faire avant de démarrer postmaster. Elles sont décrites dans les sections d'installation et de configuration de ce manuel. Cependant, si les instructions d'installation de Postgres ont été suivies à la lettre, la simple commande ci-dessous suffit à démarrer postmaster :

```
% postmaster
```

Postmaster imprime à l'occasion des messages, souvent utiles au dépannage. Si l'on souhaite avoir les messages d'aide au débogage de postmaster, le démarrer avec l'option –**d**, qui redirige la sortie vers le fichier de trace :

```
% postmaster -d >& pm.log 2>&1 &
```

Pour ne pas voir ces messages, on utilise l'option **-S**, comme « S »ilence, qui les inhibe :

```
% postmaster -S
```

Remarque : il n'y a aucun « & » à la fin du dernier exemple ; postmaster sera donc exécuté en tâche de premier plan.

© copyright Éditions Eyrolles

Les pg_options

Le fichier optionnel **data/pg_options** contient les options d'exécution utilisées par le back-end, un processus qui tourne en tâche de fond, pour contrôler les messages de trace et les autres paramètres réglables qu'il gère. Le fichier est relu par le back-end quand il reçoit un signal SIGHUP, ce qui permet à l'utilisateur de changer les options d'exécution à son gré sans avoir à redémarrer Postgres. Les options spécifiées dans ce fichier peuvent être des flags de débogage utilisés par le paquetage de trace (**backend/utils/misc/ trace.c**), ou des paramètres numériques pris en compte par le back-end pour adapter son comportement.

Toutes les **pg_options** sont initialisées à zéro au démarrage du back-end. Les options nouvelles ou celles qui sont modifiées sont lues par tous les nouveaux back-ends à leur démarrage. Pour qu'un changement soit pris en compte par tous les back-ends en fonctionnement, il faut envoyer un signal SIGHUP au postmaster, qui le renvoie automatiquement à tous les back-ends. On peut aussi activer le changement pour un back-end particulier en lui envoyant directement le SIGHUP.

Les **pg_options** peuvent aussi être stipulées avec l'option –**T** de Postgres :

```
postgres options  -T "verbose=2,query,hostlookup-"
```

Les fonctions d'impression d'erreurs et de messages de débogage peuvent maintenant utiliser le service du syslog(2). Un message imprimé sur **stdout** ou **stderr** contient la date, l'heure et aussi l'identifiant (pid) du back-end :

```
#timestamp              #pid      #message
980127.17:52:14.173  [29271]  StartTransactionCommand
980127.17:52:14.174  [29271]  ProcessUtility: drop table t;
980127.17:52:14.186  [29271]  SIIncNumEntries: table is 70% full
980127.17:52:14.186  [29286]  Async_NotifyHandler
980127.17:52:14.186  [29286]  Waking up sleeping backend process
980127.19:52:14.292  [29286]  Async_NotifyFrontEnd
980127.19:52:14.413  [29286]  Async_NotifyFrontEnd done
980127.19:52:14.466  [29286]  Async_NotifyHandler done
```

Ce format améliore la lisibilité des traces et permet de comprendre exactement ce que fait le back-end, et à quel moment. Cela facilite aussi l'écriture d'un script simple, en **awk** ou en **perl,** qui gère la trace pour détecter des erreurs ou des problèmes de base de données, ou pour calculer des statistiques sur les temps de transaction. Les messages imprimés sur **syslog** utilisent l'outil **LOG_LOCAL0** et l'option **syslog** permet de contrôler **l'**utilisation de **syslog**. Malheureusement, beaucoup de fonctions font directement appel à **printf()** pour imprimer leurs messages sur **stdout** ou **stderr**, et cette sortie ne peut être redirigée vers **syslog** ou contenir des marques horaires. Il est préfé-

© copyright Éditions Eyrolles

rable que tous les appels à **printf** soient remplacés par la macro **PRINTF** et modifier les sorties sur **stderr** de manière à utiliser **EPRINTF** : on contrôle ainsi que toutes les sorties de manière uniforme.

Le format du fichier **pg_options** est le suivant :

```
# comment
option=integer_value        # donne une valeur à l'option
option                      # met l'option à 1
option+                     # met l'option à 1
option-                     # met l'option à 0
```

Noter que **keyword** peut aussi être l'abréviation du nom d'une option définie dans **backend/utils/misc/trace.c**. Par exemple, le fichier **pg_options** peut contenir les valeurs suivantes :

```
verbose=2
query
hostlookup
showportnumber
```

Options définies

Les options définies actuellement sont décrites dans le tableau suivant :

Tableau 9–1
Options d'exécution de PostgreSQL

Options	Description
all	Flag de trace globale. Les valeurs autorisées sont : 0 messages de trace autorisés cas par cas ; 1 activer tous les messages de trace ; -1 désactiver tous les messages de trace.
verbose	Flag Commentaires. Les valeurs autorisées sont : 0 aucun message ; valeur par défaut ; 1 imprimer les messages d'information ; 2 imprimer davantage de messages d'information.
query	Flag de trace des requêtes. Les valeurs autorisées sont : 0 ne pas imprimer de requêtes ; 1 imprimer une requête condensée sur une ligne ; 4 imprimer la requête complète.
plan	Imprimer le plan de la requête.
parse	Imprimer la sortie du programme d'analyse.

© copyright Éditions Eyrolles

Tableau 9–1
Options d'exécution de PostgreSQL

Options	Description
rewritten	Imprimer la requête réécrite.
parserstats	Imprimer les statistiques du programme d'analyse.
plannerstats	Imprimer les statistiques relatives à l'agenda.
executorstats	Imprimer les statistiques d'exécution.
shortlocks	Actuellement inutilisée, mais réservée pour des caractéristiques futures.
locks	Trace des verrouillages.
userlocks	Trace des verrouillages utilisateur.
spinlocks	Trace des verrouillages par rotation.
notify	Trace des fonctions de notification.
malloc	Actuellement inutilisée.
palloc	Actuellement inutilisée.
lock_debug_oidmin	Valeur minimale de l'OID de relation tracé lors du verrouillage.
lock_debug_relid	OID, si non nul, de la trace lors du verrouillage.
lock_read_priority	Actuellement inutilisé.
deadlock_timeout	Fréquence de vérification des interblocages (deadlock).
syslog	Flag **syslog**. Les valeurs autorisées sont : 0 messages sur **stdout/stderr** ; 1 messages sur **stdout/stderr** et **syslog** ; 2 messages seulement sur **syslog**.
hostlookup	Active l'inspection des noms d'hôte dans **ps_status**.
showportnumber	Montre le numéro de port dans **ps_status**.
notifyunlock	Déverrouille **pg_listener** après notification.
notifyhack	Enlève les doublons de **pg_listener**.

© copyright Éditions Eyrolles

Sécurité

La sécurité des bases de données est traitée à plusieurs niveaux :

- Protection du fichier de base de données. Tous les fichiers stockés dans les bases de données sont protégés contre la lecture par tout compte autre que celui du super-utilisateur Postgres.
- Les connexions d'un client au serveur de base de données sont, par défaut, autorisées seulement *via* un socket UNIX local, et non des sockets TCP/IP. Le back-end doit être démarré avec l'option **–i** pour autoriser des clients à se connecter à distance.
- Les connexions des clients peuvent être restreintes à une adresse IP et/ou à un nom d'utilisateur à l'aide du fichier **pg_hba.conf**, dans **PG_DATA**.
- Les connexions des clients peuvent être authentifiées *via* d'autres paquetages externes.
- Chaque utilisateur reçoit dans Postgres un nom d'utilisateur et, en option, un mot de passe. Par défaut, les utilisateurs n'ont pas de droits d'accès en écriture aux bases de données qu'ils n'ont pas créées.
- Les utilisateurs peuvent être affectés à des groupes et les accès aux tables peuvent être restreints selon les privilèges du groupe.

Authentification de l'utilisateur

L'authentification est le processus par lequel postmaster et le serveur s'assurent que l'utilisateur qui demande l'accès aux données est bien celui qu'il prétend être. Tous les utilisateurs qui font appel à Postgres sont contrôlés par comparaison avec le contenu de la classe **pg_user**. Cependant, la vérification de l'identité réelle de l'utilisateur est assurée de différentes manières :

- Depuis le shell d'utilisateur : un back-end, démarré depuis un shell utilisateur, mémorise le user-id de l'utilisateur (réel) avant d'exécuter un **setuid** à l'user-id de l'utilisateur **postgres**. Le user-id réel sert de base au contrôle d'accès. Aucune autre authentification n'est effectuée.
- Depuis le réseau : si le système Postgres est conçu comme distribué, le port TCP Internet du processus postmaster est accessible à n'importe qui. L'administrateur configure le fichier **pg_hba.conf** dans le répertoire **PGDATA** : il spécifie le système d'authentification utilisé, selon l'hôte qui fait la connexion et la base de données à laquelle il se connecte. Voir dans **pg_hba.conf(5)** la description des systèmes d'authentification disponibles. Bien sûr, une authentification fondée sur l'hôte dans UNIX n'est pas à l'abri d'intrus qui dissimulent volontairement leur hôte d'origine. Ces problèmes de sécurité ne sont pas du ressort de Postgres.

© copyright Éditions Eyrolles

Contrôle de l'hôte

Le contrôle de l'hôte désigne les contrôles de base que PostgreSQL exerce sur les clients autorisés à accéder à une base de données, et la manière dont les utilisateurs de ces clients doivent s'authentifier. Chaque système de base de données contient un fichier nommé **pg_hba.conf**, dans son répertoire **PGDATA** : il indique qui a le droit de se connecter à une certaine base de données. Un client qui accède à une base de données doit figurer dans l'une des entrées de **pg_hba.conf**. Sinon, toutes les connexions qu'il tente sont rejetées, accompagnées du message d'erreur correspondant, qui signifie l'échec de l'authentification de l'utilisateur.

Le fichier **pg_hba.conf** est un ensemble d'enregistrements, à raison d'un par ligne. Les lignes blanches et les lignes commençant par un caractère de commentaires (#) sont ignorées. Un enregistrement est constitué de champs séparés par des espaces et/ou des tabulations.

Les connexions des clients peuvent se faire à l'aide de sockets du domaine UNIX ou de sockets du domaine Internet (c'est-à-dire, TCP/IP). Les connexions faites à l'aide des sockets UNIX sont contrôlées par des enregistrements de la forme suivante :

```
local database méthode_authentification
```

où

- **database** spécifie la base de données à laquelle cet enregistrement s'applique ; la valeur **all** qu'il s'applique à toutes les bases de données ;
- **méthode_authentification** stipule la méthode imposée aux utilisateurs pour s'authentifier lors de la connexion à cette base de données, *via* des sockets UNIX. Ces méthodes seront décrites plus loin.

Les connexions faites à l'aide de sockets Internet sont contrôlées par des enregistrements de la forme :

```
host database adresse_TCP/IP masque_TCP/IP méthode_authentification
```

L'**adresse TCP/IP** fait l'objet d'un ET logique avec le masque TCP/IP spécifié et l'adresse TCP/IP du client qui se connecte. Si les deux résultats sont égaux, l'enregistrement est retenu pour la connexion. Si une connexion concorde avec plusieurs enregistrements, c'est le premier trouvé dans le fichier qui est utilisé. L'adresse et le masque TCP/IP sont spécifiés dans la notation avec des points décimaux. Si la connexion ne concorde avec aucun des enregistrements, c'est la méthode de rejet de l'authentification qui s'applique (voir plus loin).

© copyright Éditions Eyrolles

Méthodes d'authentification

Les méthodes d'authentification suivantes valent aussi bien pour les sockets du domaine TCP/IP que pour ceux du domaine UNIX :

- **trust** : la connexion est autorisée sans condition.
- **reject** : la connexion est rejetée sans condition.
- **crypt** : le client doit fournir le mot de passe de l'utilisateur, qui est envoyé crypté (à l'aide de **crypt(3)**) et comparé au mot de passe contenu dans la table **pg_shadow.** S'ils concordent, la connexion est autorisée.
- **password** : le client doit fournir le mot de passe de l'utilisateur. Il est envoyé en clair et comparé au mot de passe contenu dans la table **pg_shadow.** Si les mots de passe concordent, la connexion est autorisée. Un mot de passe peut être spécifié après le mot-clé **password** : il sert à vérifier le mot de passe au lieu de la table **pg_shadow.** Voir **pg_passwd**.

Les méthodes d'authentification suivantes ne valent que pour les sockets du domaine TCP/IP :

- **krb4** : Kerberos V4 est utilisé pour authentifier l'utilisateur.
- **krb5** : Kerberos V5 est utilisé pour authentifier l'utilisateur.
- **ident** : l'identité du serveur sur le client sert à authentifier l'utilisateur (RFC 1413). Un nom de map optionnel peut être spécifié après le mot-clé **ident** : il permet de relier les noms d'utilisateurs de ident à leurs noms d'utilisateurs dans Postgres. Les correspondances sont gérées dans le fichier **$PGDATA/pg_ident.conf**.

Voici quelques exemples :

```
# Toute connection via les sockets Unix : trust
local trust
# Toute connection via les TCP/IP de cette machine : trus
host all 127.0.0.1 255.255.255.255 trust
# On ne sait rien de cette machine : reject
host all 192.168.0.10 255.255.255.0 reject
# Celle-ci ne peut pas crypter: on demande les mots de passe en clair.
host all 192.168.0.3 255.255.255.0 password
# Les autres machines peuvent fournir des mots de passe cryptés.
host all 192.168.0.0 255.255.255.0 crypt
```

Authentification Kerberos

Kerberos est un système d'authentification sécurisé au standard des professionnels et adapté aux réseaux informatiques publics.

© copyright Éditions Eyrolles

Disponibilité

Le système d'authentification Kerberos n'est pas distribué par Postgres. Les versions de Kerberos sont en général fournies en option par les éditeurs de système d'exploitation. De plus, une distribution du code source peut être obtenue depuis Athena, projet du MIT.

Important. Vous pouvez obtenir la version MIT même si votre éditeur fournit une version de Kerberos, car certains portages des éditeurs ont été délibérément dénaturés ou rendus incompatibles avec la version du MIT.

Les utilisateurs situés en dehors des États-Unis et du Canada doivent savoir que la distribution du code de cryptage réel de Kerberos est restreinte de par la réglementation à l'export fixée par le gouvernement américain. Les requêtes concernant la version de Kerberos que vous souhaitez doivent être adressées à votre éditeur ou au projet Athena du MIT. Notez que des listes de FAQ sont postées périodiquement sur la liste de diffusion de Kerberos (envoyer un message pour y souscrire), et sur le groupe de news USENET.

Installation

Les notices de Kerberos décrivent en détail son installation. S'assurer que le fichier de clé du serveur (**srvtab** ou **keytab**) est d'une manière ou d'une autre accessible au compte Postgres.

Postgres et ses clients peuvent être compilés pour utiliser la version 4 ou la version 5 des protocoles Kerberos du MIT — mettre la valeur appropriée dans la variable KRBVERS du fichier **src/Makefile.global**. On peut aussi changer l'emplacement où Postgres s'attend à trouver les bibliothèques associées, les fichiers d'en-tête et son propre fichier de clés du serveur. Après compilation, Postgres doit être enregistré en tant que service Kerberos : voir les modes opératoires de Kerberos et les pages de manuel associées pour plus de détails sur l'enregistrement de services.

Mise en œuvre

Après l'installation initiale, Postgres fonctionnera dans tous les cas comme un service Kerberos normal. Pour plus de détails sur l'utilisation de l'authentification, voir dans le *Guide de l'utilisateur PostgreSQL* les sections relatives à **postmaster** et à **psql**.

Dans la documentation de la version 5 de Kerberos, on suppose, à propos de la dénomination des utilisateurs et des services (voir tableau 9.1) que :

- les noms principaux des utilisateurs (*anames*) ne contiennent pas de nom réel d'utilisateur UNIX/Postgres dans le premier composant ;

© copyright Éditions Eyrolles

Tableau 9–1
Exemples de paramètres Kerberos

Paramètre	Exemple
utilisateur	frew@S2K.ORG
utilisateur	aoki/HOST=miyu.S2K.Berkeley.EDU@S2K.ORG
hôte	postgres_dbms/ucbvax@S2K.ORG

- le service Postgres contient deux composants : le nom de service et un nom d'hôte, « générique » comme en version 4 (tous les suffixes de domaines sont supprimés).

Le support de la version 4 est appelé à disparaître quand le MIT aura publié la version 5 de production.

Créer des utilisateurs

Pour définir un nouvel utilisateur, on exécute l'utilitaire **createuser.** Pour assigner un utilisateur ou un ensemble d'utilisateurs à un nouveau groupe, il faut définir le groupe puis lui affecter les utilisateurs. Dans Postgres, il n'existe pas encore de commande **create group.** : **l**es groupes sont donc définis en ajoutant les valeurs appropriées dans la table système **pg_group** et en se servant de la commande **grant** pour leur affecter des privilèges.

Créer des groupes

Actuellement, il n'y a pas d'interface simple destinée à initialiser les groupes d'utilisateurs. On doit explicitement insérer ou mettre à jour un enregistrement de la table **pg_group.** Par exemple :

```
jolly=> insert into pg_group (groname, grosysid, grolist)
jolly=>     values ('posthackers', '1234', '{5443, 8261}');
    INSERT 548224
jolly=> grant insert on foo to group posthackers;
   CHANGE
jolly=>
```

Les champs de **pg_group** sont :

- **groname** : nom du groupe. Ce nom sera strictement alphanumérique — sans caractère de soulignement ni de ponctuation.

© copyright Éditions Eyrolles

- **grosysid** : id du groupe, de type int4, unique pour chaque groupe.
- **grolist** : liste des ID de **pg_user** qui appartiennent au groupe. C'est un champ de type int4[ ].

Ajouter et supprimer des utilisateurs

La commande **createuser** permet de créer des utilisateurs spécifiques qui accèdent à Postgres, et **destroyuser** les supprime ainsi que leur droit d'accès à Postgres. Ces commandes n'affectent que les droits de l'utilisateur dans Postgres ; elles n'ont aucun effet sur les autres privilèges de l'utilisateur ni sur son statut vis-à-vis du système d'exploitation sous-jacent. Voici un exemple de création :

```
bash# su - postgres
bash$ man createuser
bash$ createuser
```

Contrôle des accès

Postgres donne à l'utilisateur les moyens de limiter l'accès à ses données. Voici quelques-unes de ces possibilités :

- super-utilisateurs de bases de données (**database superusers**), c'est-à-dire les utilisateurs habilités par **pg_user.usesuper** : ils contournent, sans même que l'on s'en aperçoive, tous les contrôles d'accès sauf dans deux cas, la mise à jour manuelle des catalogues système, autorisée seulement si l'utilisateur a le droit **pg_user.usecatupd,** et la destruction de catalogues système (ou la modification de leurs schémas), jamais autorisée.
- privilège d'accès (**access privilege**) : l'utilisation des privilèges d'accès pour limiter la lecture, l'écriture et la mise à jour des règles sur les classes est traitée par **grant/revoke(1)**.
- suppression de classe et modification de schéma (**class removal and schema modification**) : les commandes qui détruisent ou modifient la structure d'une classe existante, telles alter, drop table et drop index ne fonctionnent que pour le propriétaire. Pour rappel, ces opérations ne sont jamais permises sur les catalogues système.

© copyright Éditions Eyrolles

Fonctions et règles

Les fonctions et les règles permettent à l'utilisateur d'insérer, dans le serveur, du code que les autres utilisateurs peuvent exécuter sans le savoir, ce qui donne une relative impunité aux utilisateurs qui introduisent des chevaux de Troie. La seule protection réelle est d'effectuer un contrôle serré du droit de définir des fonctions (par exemple, se servir de champs SQL pour enregistrer des relations) et des règles. On peut aussi tirer parti des traces d'audit et des alertes sur **pg_class**, **pg_user** et **pg_group**.

Les fonctions écrites en n'importe quel langage, sauf SQL, s'exécutent dans le serveur avec les permissions de l'utilisateur Postgres (le serveur lui-même fonctionne avec son user-id réel et ses droits attribués dans Postgres). Les utilisateurs peuvent donc changer les structures internes des données du serveur à travers des fonctions autorisées. C'est pourquoi, entre autres, de telles fonctions peuvent contourner tout contrôle d'accès du système. C'est un problème inhérent aux fonctions C définies par l'utilisateur.

Tout comme les fonctions SQL, les règles fonctionnent toujours sous l'identité et les permissions de l'utilisateur qui a fait appel au serveur.

Mises en garde

Il n'existe pas de méthode qui gère l'encryptage des données dans Postgres (bien qu'il n'y ait rien qui interdise à l'utilisateur de crypter ses données dans des fonctions définies par l'utilisateur). De même, il n'y a aucune méthode pour gérer les connexions réseau cryptées, sauf à attendre une réécriture complète du protocole front-end/back-end.

Les noms d'utilisateurs, les noms de groupe et les identifiants système associés (par exemple, le contenu de **pg_user.usesysid**) sont supposés être uniques au sein d'une base de données. Si ce n'est pas le cas, il faut s'attendre à des résultats imprévisibles.

Sécuriser les connexions TCP/IP

On peut utiliser **ssh** pour crypter la connexion réseau entre un serveur Postgres et les clients. Si l'on s'y prend bien, on établit les conditions d'une connexion réseau sûre. La documentation de ssh donne la plupart des informations nécessaires au démarrage. Pour plus de détails, se reporter à la page http://www.heimhardt.de/htdocs/ssh.html.

© copyright Éditions Eyrolles

Exécuter une passerelle sécurisée *via* ssh

Voici un cas, traité en deux étapes seulement.

- Établir une passerelle vers le serveur, comme ceci :

```
ssh -L 3333:wit.mcs.anl.gov:5432 postgres@wit.mcs.anl.gov
```

  Le premier nombre, **3333,** dans l'argument **-L**, est le numéro de port de l'extrémité locale de la passerelle. Le second, **5432**, représente l'extrémité distante de la passerelle — le numéro de port dont se sert le serveur. Le nom, ou l'adresse, qui prend place entre les numéros de port appartient à la machine serveur, ainsi que le dernier argument de **ssh**, qui contient un nom d'utilisateur optionnel. En l'absence de ce dernier, **ssh** essaie d'utiliser le nom sous lequel on est actuellement connecté sur la machine cliente. On peut utiliser tout nom d'utilisateur que la machine serveur est susceptible d'accepter, pas nécessairement celui qui se réfère Postgres.

- Maintenant que l'on a une session **ssh** qui fonctionne : connecter un client Postgres à l'hôte local sur le numéro de port spécifié à l'étape précédente. Si c'est **psql**, ouvrir une autre session de shell parce que celle de l'étape 1 est maintenant prise par **ssh**.

```
psql -h localhost -p 3333 -d mpw
```

  Noter que l'on doit spécifier l'argument **-h** pour que le client utilise le socket TCP plutôt que le socket UNIX. On peut omettre l'argument de port si l'on a **5432** comme extrémité de la passerelle.

Gestion des disques et support des grandes bases de données

PostgreSQL peut gérer des bases de données extrêmement grandes, dont la taille dépasse 200 Go ! Les performances des machines 32 bits déclinent rapidement, dès que la taille de la base de données excède 5 Go. On arrive à faire fonctionner une base de données de 30 Go sur un processeur 32 bits, mais les performances sont dégradées. Les processeurs 32 bits imposent des limites de 2 Go sur la RAM (bien que les nouvelles versions de Linux aient étendu quelque peu cette limite), de 2 Go sur la taille du système de fichiers, et d'autres limitations sur le système d'exploitation. Il faut utiliser les systèmes de fichiers spécialement conçus pour Linux par SGI, IBM ou HP, ou le système ext3-fs pour supporter des tailles de fichiers supérieures à 2 Go sur les machines Linux 32 bits.

© copyright Éditions Eyrolles

Pour les bases de données extrêmement grandes, il est vivement conseillé d'utiliser des machines à processeurs 64 bits, telles Digital Alpha, Sun UltraSPARC 64 bits, Silicon Graphics 64 bits, Intel Merced IA-64, les machines 64 bits HPUX ou IBM. Compiler PostgreSQL pour un processeur 64 bits, et il peut supporter d'immenses bases de données et des requêtes de grande envergure. Les performances de PostgreSQL pour les requêtes sur de grandes tables ou bases de données seront plusieurs fois supérieures à celles de PostgreSQL sur les machines 32 bits. Les machines 64 bits ont l'avantage de disposer d'un très grand espace d'adressage mémoire, de sorte que le système d'exploitation peut supporter de très grands systèmes de fichiers, fournir de meilleures performances avec les grandes bases de données et disposer de beaucoup plus de mémoire pour améliorer l'efficacité de ses fonctionnalités.

Localisation secondaire

Il est possible de créer une base de données à un emplacement autre que l'emplacement par défaut de l'installation. Se souvenir que tout accès à une base de données se produit réellement *via* le back-end de la base de données, de sorte que tout emplacement spécifié doit être accessible par le back-end.

Les emplacements secondaires de bases de données sont créés et référencés par une variable d'environnement, qui donne le chemin absolu de l'emplacement de stockage. Elle doit avoir été définie avant le démarrage du back-end, et pouvoir être écrite depuis le compte de l'administrateur de Postgres. Tout nom de variable d'environnement valide peut servir à référencer un emplacement, bien que l'usage d'un nom de variable préfixé par PGDATA soit recommandé, pour éviter la confusion et les conflits avec d'autres variables.

Important. Dans les versions précédentes de Postgres, il était possible d'utiliser un nom de chemin absolu pour spécifier un emplacement de stockage secondaire. La spécification à l'aide d'une variable d'environnement est préférable, car elle donne plus de souplesse à l'administrateur pour gérer l'espace disque. Si l'on préfère se servir de chemins absolus, redéfinir ALLOW_ABSOLUTE_DBPATHS et recompiler Postgres. Pour ce faire, on peut soit ajouter la ligne

```
#define ALLOW_ABSOLUTE_DBPATHS 1
```

au fichier **src/include/config.h**, soit ajouter la spécification

```
CFLAGS+= -DALLOW_ABSOLUTE_DBPATHS
```

au fichier **makefile.custom**.

sachant que la création de la base de données est exécutée réellement par le back-end de la base de données. Par conséquent, toute variable d'environnement qui spécifie un emplacement secondaire doit avoir été définie avant le

© copyright Éditions Eyrolles

démarrage du serveur. Pour définir un emplacement secondaire, **PGDATA2**, qui pointe vers **/home/postgres/data**, entrer :

```
% setenv PGDATA2 /home/postgres/data
```

qui définit la variable d'environnement à utiliser dans les commandes suivantes. En général, on définit cette variable dans le fichier **.profile** ou le fichier d'initialisation **.cshrc** du super-utilisateur Postgres pour s'assurer qu'elle est bien renseignée au démarrage du système. Toute variable d'environnement peut référencer un emplacement secondaire, bien qu'il soit préférable d'utiliser des variables préfixées par **PGDATA** pour éliminer toute confusion et tout risque d'incompatibilité ou d'écrasement des autres variables.

Pour créer une zone de stockage de données dans **PGDATA2**, s'assurer que **/home/postgres** existe déjà et que l'administrateur Postgres peut écrire dedans. Dans la ligne de commande, taper : :

```
% setenv PGDATA2 /home/postgres/data
% initlocation $PGDATA2
Creating Postgres database system directory /home/postgres/data

Creating Postgres database system directory /home/postgres/data/base
```

Pour tester le nouvel emplacement, créer une base de données de test à l'aide de la commande :

```
% createdb -D PGDATA2 test
% dropdb test
```

Gérer une base de données

Si le postmaster Postgres est démarré, nous pouvons créer quelques bases de données pour l'expérimenter. Ici, nous décrivons les principales commandes qui gèrent les bases de données.

Créer une base de données

Pour créer une base de données nommée **mydb**, par exemple, entrer la commande suivante :

```
% createdb dbname
```

© copyright Éditions Eyrolles

Postgres permet de créer un nombre quelconque de bases de données sur un site donné, et l'on devient automatiquement administrateur de la base de données que l'on vient de créer. Les noms de bases de données commencent obligatoirement par un caractère alphabétique et sont limités à une longueur de 31 caractères. Tous les utilisateurs n'ont pas l'autorisation de devenir administrateur de base de données : si Postgres refuse de vous créer une base de données, c'est à l'administrateur du site de vous accorder cette permission. Consulter votre administrateur de site si cela se produit.

Accéder à une base de données

Une fois que l'on a construit une base de données, on peut y accéder de deux façons :

- à l'aide du programme terminal de Postgres (psql), qui permet d'entrer, d'éditer et d'exécuter des commandes SQL de manière interactive ;
- en écrivant un programme C qui se sert de la bibliothèque **libpq**, pour soumettre des commandes SQL depuis le C et obtenir des réponses et des messages d'état en retour dans le programme. Cette interface est décrite plus loin dans le guide du programmeur PostgreSQL.

Commencer par **psql** et essayer les exemples du manuel. On l'active sur une base de données **dbname** par la commande :

```
% psql dbname
```

On est accueilli par le message suivant :

```
Welcome to the Postgres interactive sql monitor :

  type \? for help on slash commands
  type \q to quit
  type \g or terminate with semicolon to execute query
You are currently connected to the database: dbname

dbname=>
```

Cette invite indique que le terminal est à l'écoute et que l'on peut entrer des requêtes SQL dans l'espace de travail du terminal. Le programme **psql** réagit aux codes d'échappement qui commencent par le caractère « \ » (barre inversée). Par exemple, on obtient l'aide sur la syntaxe des diverses commandes SQL de Postgres en tapant :

```
dbname=> \h
```

© copyright Éditions Eyrolles

Après avoir entré une requête dans l'espace de travail, on envoie le contenu de l'espace de travail au serveur Postgres en entrant :

```
dbname=> \g
```

Le serveur va alors traiter la requête. Si l'on termine la requête par un point-virgule, le **\g** n'est pas nécessaire. **psql** traite automatiquement les requêtes terminées par un point-virgule. Pour lire des requêtes depuis un fichier, au lieu de les taper, entrer :

```
dbname=> \i filename
```

Pour sortir de **psql** et revenir à UNIX, taper

```
dbname=> \q
```

psql s'arrête et l'on revient au shell de commande. (Pour connaître d'autres codes d'échappement, taper **\-h** à l'invite du terminal.) Les blancs — espaces, tabulations et retours à la ligne — peuvent être utilisés librement dans les requêtes SQL. Les commentaires sur une seule ligne sont signalés par deux tirets (--) : tout ce qui suit les tirets, jusqu'à la fin de la ligne, est ignoré. Les commentaires sur plusieurs lignes et les commentaires dans une ligne sont indiqués par /* ... */, une convention empruntée à Ingres.

Détruire une base de données

Si l'on est administrateur de la base de données **mydb**, on peut la détruire à l'aide de la commande UNIX suivante :

```
% dropdb dbname
```

Cette action efface physiquement tous les fichiers UNIX associés à la base de données. Elle ne peut être annulée, aussi doit-on l'utiliser avec une grande prudence. On peut aussi détruire une base de données dans une session SQL à l'aide de la commande :

```
> drop database dbname
```

Sauvegarde et restauration

Postgres met à disposition deux utilitaires de sauvegarde du système : **pg_dump** pour la sauvegarde d'une seule base de données et **pg_dumpall** pour la sauvegarde de toute l'installation en une seule étape.

© copyright Éditions Eyrolles

Une base de données peut être sauvegardée isolément par la commande :

```
% pg_dump dbname  > dbname .pgdump
```

et restaurée par :

```
cat dbname.pgdump | psql dbname
```

Cette technique sert encore à transférer des bases de données vers de nouveaux emplacements ou à renommer des bases de données existantes.

Attention. Toute base de données doit être sauvegardée dans les règles. Postgres gère ses propres fichiers dans le système de fichiers, il n'est donc pas recommandé de compter sur le système de sauvegarde du système de fichiers pour assurer la sauvegarde d'une base de données : dans ce cas, la restauration ne serait pas fiable.

Grandes bases de données

Depuis que Postgres admet des tables d'une taille plus grande que la taille maximale des fichiers sur le système, on peut rencontrer des problèmes lors du dump d'une table dans un fichier, car le fichier résultat risque de dépasser la taille maximale autorisée sur le système. Quand **pg_dump** écrit sur **stdout**, il est de bon aloi d'utiliser les outils *nix standard pour contourner ce problème :

- Utiliser un dump compressé :

```
% pg_dump dbname  | gzip > filename .dump.gz
```

et recharger avec

```
% createdb dbname
% gunzip -c filename.dump.gz | psql dbname
```

ou bien

```
% cat filename.dump.gz | gunzip | psql dbname
```

- Utiliser l'option split :

```
% pg_dump dbname | split -b 1m - filename.dump.
```

et recharger avec

```
% createdb dbname
% cat filename.dump.* | pgsql dbname
```

© copyright Éditions Eyrolles

Bien sûr, le nom du fichier (**filename**) et celui de la sortie de **pg_dump** n'ont pas besoin d'être cohérents avec le nom de la base de données. La base de données restaurée peut avoir un nouveau nom, arbitraire : ce mécanisme peut donc aussi servir à renommer des bases de données.

Utiliser un boîtier de commutation avec PostgreSQL

On peut empiler plusieurs serveurs et les connecter à un même terminal en passant par un boîtier de commutation KVM (Keyboard, Vidéo, Monitor). Cela évite l'encombrement, économise de l'espace, et aussi de l'argent : le moniteur, le clavier et la souris coûtent de 100 à 500 dollars US par ensemble. Avec ce boîtier de commutation, on peut empiler beaucoup de serveurs dans une tour : PostgreSQL (développement, test, production), web, ftp, intranet, serveurs de courrier et de nouvelles. Le boîtier de commutation peut servir aussi à contrôler Windows 95/NT, ainsi que OS/2.

Consultez les sites suivants :

- DataComm Warehouse Inc. Téléphone 1–800–328–2261
  http://www.warehouse.com.
  fournit toutes sortes de matériels informatiques : un boîtier KVM manuel à quatre ports (PS/2) coûte 89.99 dollars (Référence : DDS1354).
- Network Technologies Inc.
  http://www.networktechinc.com/servswt.html.
  propose des commutateurs de serveurs et des commutateurs vidéo (120 dollars par PC 8 ports).
- Scene Double Inc., Angleterre
  http://www.scene.demon.co.uk/qswitch.htm.
- Cybex Corporation : http://www.cybex.com.
- Raritan Inc. : http://www.raritan.com.
- RealStar Solutions Inc. : http://www.real-star.com/kvm.htm.
- Belkin Inc. : http://www.belkin.com.
- Better Box Communications Ltd.
  http://www.betterbox.com/info.html.

Rendez-vous au magasin de matériel le plus proche et demandez un commutateur de serveurs, connu aussi sous le nom de commutateur KVM automatique. Pensez aussi à utiliser le moteur de recherche Yahoo! pour trouver d'autres sociétés qui proposent des commutateurs de serveur ou des commutateurs KVM.

© copyright Éditions Eyrolles

Il est préférable d'avoir une machine UNIX dédiée à chaque serveur de base de données PostgreSQL, pour obtenir de meilleures performances : aucune autre application ne doit fonctionner sur cette machine. On n'a pas besoin d'un moniteur couleur pour le serveur de base de données, car on peut l'administrer à distance à partir d'une station de travail PC couleur. Se servir d'un commutateur KVM permet de contrôler de nombreuses machines avec un seul moniteur et un seul clavier !

Échec au démarrage de postmaster

Certaines pannes au démarrage de postmaster sont courantes et faciles à comprendre. Vérifier le fichier de trace de postmaster ou le démarrer à la main (sans rediriger ni la sortie standard ni la sortie des erreurs) pour analyser les messages émis. Certains messages d'erreur sont évidents et s'expliquent d'eux-mêmes, mais en voici quelques-uns qui ne le sont pas :

```
FATAL : StreamServerPort : bind() failed : Address already in use
        Is another postmaster already running on that port ?
```

En général, celui-ci est à prendre au premier degré : n'auriez-vous pas, par hasard, démarré un second postmaster sur un port où un autre postmaster est déjà actif ? Cependant, si le corps du message d'erreur n'est pas « Adresse déjà utilisée » ou une variante de ce libellé, il y a peut-être un autre problème. Par exemple, essayer de démarrer un postmaster sur un numéro de port réservé donne quelque chose comme :

```
$ postmaster -i -p 666
FATAL: StreamServerPort: bind() failed: Permission denied
        Is another postmaster already running on that port?

IpcMemoryCreate: shmget failed (Invalid argument) key=5440001,
  size=83918612, permission=600
FATAL 1 :  ShmemCreate : cannot create region
```

Ce message signifie probablement que la taille limite des zones de mémoire partagée du noyau est inférieure à la zone de buffer que Postgres essaie de créer — cela peut signifier aussi que le noyau n'est pas du tout configuré avec le support de mémoire partagée de type SysV. Un moyen de contournement provisoire : démarrer le postmaster avec un nombre de buffers plus petit que la normale (option **–B**). Il faudra certainement reconfigurer le noyau pour augmenter la taille de mémoire partagée autorisée. Le message suivant apparaît si l'on essaie de démarrer plusieurs postmasters sur la même machine, et que leur besoin total d'espace excède la limite du noyau.

```
IpcSemaphoreCreate : semget failed (No space left on device) key=5440026,
num=16, permission=600
```

© copyright Éditions Eyrolles

Ce message ne signifie pas que l'on a saturé l'espace disque, mais que la limite du nombre de sémaphores SysV du noyau est inférieure au nombre de sémaphores que Postgres veut créer. Comme ci-dessus, on peut contourner le problème en démarrant postmaster avec un nombre réduit de processus de back-end (option **–N**), mais il faudra augmenter la limite du noyau.

Problèmes de connexion du client

Une fois que postmaster fonctionne, la connexion d'applications clientes peut encore échouer pour diverses raisons. Les exemples de messages d'erreur qui suivent ne concernent que les clients s'appuyant sur les versions récentes de **libpq** ; les clients qui font appel à d'autres bibliothèques d'interface peuvent produire d'autres messages, plus ou moins explicites.

```
connectDB() – connect() failed : Connection refused
Is the postmaster running (with -i) at 'server.joe.com' and accepting
connections on TCP/IP port '5432' ?
```

C'est l'échec générique : « Je ne trouve pas de postmaster à qui parler ».

L'exemple précédent concerne le cas d'une communication TCP/IP, et le suivant, celui d'une communication locale via un socket UNIX :

```
connectDB() – connect() failed : No such file or directory
Is the postmaster running at 'localhost' and accepting connections
on Unix socket '5432' ?
```

La dernière ligne sert à vérifier à quoi le client essaie de se connecter. S'il n'y a aucun postmaster en fonctionnement, le corps du message d'erreur sera « Connexion refusée » ou « Fichier ou répertoire inconnu » (*No such file or directory*), comme dans l'exemple. Remarque : il faut bien noter que « Connexion refusée » dans ce contexte ne signifie pas que le postmaster a rejeté la demande de connexion — ce cas produira un message différent, que nous allons voir. D'autres messages d'erreur, tel « Temps de connexion écoulé » (*Connection timed out*), traduisent des problèmes plus fondamentaux, comme l'absence de connectivité du réseau.

```
No pg_hba.conf entry for host 123.123.123.123, user joeblow, database testdb
```

On tente de se connecter à un postmaster, sachant qu'il risque de refuser la connexion. Comme l'indique le message, postmaster refuse la demande de connexion parce qu'il n'a trouvé aucune entrée dans son fichier de configuration **pg_hba.conf.**

```
Password authentication failed for user 'joeblow'
```

© copyright Éditions Eyrolles

Ce message indique que le postmaster n'accepte pas la demande de connexion tant que l'on n'aura pas satisfait à la méthode d'autorisation spécifiée dans **pg_hba.conf.** Vérifier le mot de passe fourni, ou bien les logiciels Kerberos ou IDENT, si l'un de ces types d'authentification est en cause.

```
FATAL 1 :  SetUserId : user 'joeblow' is not in 'pg_shadow'
```

Il s'agit d'une autre variante d'échec de l'authentification : aucune commande Postgres **create_user** n'a été exécutée avec le nom d'utilisateur donné.

```
FATAL 1 :  Database testdb does not exist in pg_database
```

Le nom de la base de données est inconnu du postmaster. Noter que si l'on ne spécifie pas le nom d'une base de données, c'est le nom par défaut associé à l'utilisateur Postgres qui est pris en compte, ce qui n'est pas forcément bon.

Messages de débogage

Il arrive que postmaster imprime des messages souvent utiles au dépannage. Pour voir les messages de débogage du postmaster, le démarrer avec l'option –**d** et rediriger la sortie vers un fichier de trace :

```
% postmaster -d > pm.log 2>&1 &
```

Pour ne pas les voir, entrer

```
% postmaster -S
```

et postmaster restera « S »ilencieux. Remarque : il n'y a pas de « & » à la fin du dernier exemple ; postmaster sera donc exécuté en tâche de premier plan.

pg_options

Le fichier **data/pg_options** contient les options d'exécution utilisées par le back-end pour contrôler les messages de trace et d'autres paramètres réglables du back-end. L'intérêt de ce fichier est d'être relu par un back-end quand il reçoit un signal SIGHUP, ce qui permet de changer les options d'exécution à la demande sans redémarrer Postgres. Les options spécifiées dans ce fichier peuvent être des flags de débogage, utilisés par le paquetage de trace (**backend/utils/misc/trace.c**), ou des paramètres numériques utilisés directement par le back-end pour adapter son comportement. Les options ou paramètres nouveaux doivent être inclus dans les fichiers **backend/utils/misc/trace.c** et **backend/include/utils/trace.h**.

© copyright Éditions Eyrolles

Les **pg_options** peuvent aussi être stipulées avec l'option –**T** de Postgres :

```
postgres options  -T "verbose=2,query,hostlookup-"
```

Les fonctions d'impression d'erreurs et de messages de débogage peuvent maintenant passer par **syslog(2)**. Un message imprimé sur **stdout** ou **stderr** contient la date, l'heure et l'identifiant (pid) du back-end :

```
#timestamp            #pid      #message
980127.17:52:14.173  [29271]  StartTransactionCommand
980127.17:52:14.174  [29271]  ProcessUtility: drop table t;
980127.17:52:14.186  [29271]  SIIncNumEntries: table is 70% full
980127.17:52:14.186  [29286]  Async_NotifyHandler
980127.17:52:14.186  [29286]  Waking up sleeping backend process
980127.19:52:14.292  [29286]  Async_NotifyFrontEnd
980127.19:52:14.413  [29286]  Async_NotifyFrontEnd done
980127.19:52:14.466  [29286]  Async_NotifyHandler done
```

Ce format améliore la lisibilité des traces et permet de comprendre exactement ce que fait le back-end, et à quel moment. Cela facilite aussi l'écriture de scripts simples, en **awk** ou **Perl,** qui gèrent la trace pour détecter des erreurs ou des problèmes de base de données, ou pour calculer des statistiques sur les temps de transaction. Les messages imprimés dans **syslog** utilisent **LOG_LOCAL0** et l'option **syslog** permet de contrôler l'utilisation de **syslog**. Malheureusement, beaucoup de fonctions font directement appel à **printf()** pour imprimer leurs messages sur **stdout** ou **stderr**, et cette sortie ne peut être redirigée vers **syslog** ou contenir des marques horaires. Il vaut mieux remplacer tous les appels à **printf** par la macro **PRINTF** et modifier les sorties sur **stderr** de manière à utiliser **EPRINTF** : on peut alors contrôler l'ensemble des sorties de manière uniforme.

Le format du fichier **pg_options** est le suivant :

```
# comment
option=integer_value        # donne une valeur à l'option
option                      # met l'option à 1
option+                     # met l'option à 1
option-                     # met l'option à 0
```

keyword peut être aussi l'abréviation du nom d'une option définie dans **backend/utils/misc/trace.c**. Voir le tableau de la section « Les pg_options », page 389, pour la liste complète des options et des valeurs possibles.

© copyright Éditions Eyrolles

Ressources PostgreSQL sur l'Internet

De nombreuses ressources de l'Internet sont consacrées à PostgreSQL. Nous allons trouver, successivement :

- le support technique ;
- les listes de diffusion ;
- des outils graphiques pour PostgreSQL ;
- les pilotes ODBC, JDBC et UDBC ;
- les interfaces Perl et DBI ;
- des didacticiels pour PostgreSQL ;
- les sites de référence de PostgreSQL.

Support technique

L'organisation PostgreSQL vend des prestations d'assistance technique aux sociétés et le revenu généré sert à soutenir les principaux sites web et un certain nombre de sites miroir (web et ftp) dans le monde. Il sert aussi à financer la Recherche & Développement, ainsi qu'à la production de la documentation imprimée, des guides et manuels destinés aux clients.

On peut aussi obtenir une assistance auprès de sociétés de conseil professionnelles telles que Red Hat, Anderson[2], et WGS (*Work Group Solutions*). Ils ont une très bonne expertise en C et C++ (PostgreSQL est écrit en C).

Pour les contacter :

- Red Hat Corp, division consulting base de données : http://www.redhat.com
- Work Group Solutions : http://www.wgs.com
- Andersen Consulting : http://www.ac.com

On peut aussi utiliser les listes de diffusion et les groupes de news Usenet. Pour toute question technique, ou autre problème, envoyer un e-mail à :

- pgsql-questions@postgresql.org
- groupe de news : comp.databases.postgresql.general
- groupe de news : comp.databases.postgresql.hackers
- groupe de news : comp.databases.postgresql.doc

2. L'auteur mentionne Anderson, mais il semble qu'il s'agisse d'Andersen Consulting : c'est, en tous cas, le nom qui apparaît à l'adresse indiquée ensuite - www.ac.com. *N.d.T*

© copyright Éditions Eyrolles

- groupe de news : comp.databases.postgresql.bugs
- groupe de news : linux.postgres
- autre mailing listes à http://www.postgresql.org

En principe, le délai de réponse à un e-mail est inférieur à une journée. Comme les utilisateurs des produits Internet représentent un ensemble très vaste, et que les utilisateurs correspondent et s'assistent les uns les autres, l'Internet pourra facilement apporter de l'assistance technique à des milliards d'individus. L'e-mail est beaucoup plus commode que le téléphone, parce que l'on peut copier-coller des messages d'erreur, des sorties de programme, etc. et les envoyer sur une liste de diffusion ou à un groupe de news.

Listes de diffusion

Voici une série de listes de diffusion PostgreSQL auxquelles on peut faire appel pour obtenir à la fois des nouvelles, des informations et des réponses à des questions en rapport avec PostgreSQL.

Comptes e-mail pour PostgreSQL

Les sources suivantes donnent accès à des comptes e-mail gratuits :

- dans Yahoo! : http://www.yahoo.com,
  cliquer sur Yahoo! Mail et suivre la procédure d'inscription
- dans Lycos : http://www.lycos.com,
  cliquer sur E-mail gratuit, dans la rubrique Services, et suivre la procédure d'inscription.
- dans hotmail : http://www.hotmail.com,
  suivre la procédure d'inscription pour un nouveau compte e-mail.

Souscrire à la liste de diffusion PostgreSQL. Yahoo! propose, en supplément, de créer un dossier à part pour les messages PostgreSQL, de sorte que la boîte e-mail normale ne soit pas saturée. Dans le menu E-mail | Options | Filters, opter pour un dossier séparé pour l'e-mail. Ce compte est accessible depuis n'importe quel poste dans le monde, dès lors que l'on a accès à une page web.

Avec tout autre e-mail, on peut utiliser une option « Filtrer le courrier » pour recevoir automatiquement les messages PostgreSQL, dans un dossier séparé.

Listes de diffusion en anglais

On trouve les listes de diffusion sur la page web principale de :

http://www.postgresql.org/

© copyright Éditions Eyrolles

Pour poser une question par e-mail :

- pgsql-questions@postgresql.org, pour des questions générales
- pgsql-hackers@postgresql.org, pour les développeurs
- pgsql-ports@postgresql.org, pour les questions spécifiques d'une version
- pgsql-docs@postgresql.org, pour les questions de documentation

On obtient une réponse en moins d'une journée.

On peut aussi s'inscrire à des listes de diffusion. Pour s'inscrire à la liste ou annuler l'inscription, envoyer un message à :

- pgsql-questions-request@postgresql.org
- pgsql-hackers-request@postgresql.org
- pgsql-ports-request@postgresql.org
- pgsql-docs-request@postgresql.org

Le corps du message doit simplement contenir **subscribe** ou **unsubscribe**.

Archives de listes de diffusion

Les listes de diffusion sont aussi archivées en format HTML aux emplacements suivants :

- Des listes datées, disponibles *via* Mhonarc, à l'adresse http://www.postgresql.org/mhonarc/pgsql-questions
- Le répertoire d'archives sur ftp://ftp.postgresql.org : **/pub/majordomo**

Il y a aussi un moteur de recherche disponible sur le site web principal de PostgreSQL, en particulier pour les questions sur **pgsql**.

Listes de diffusion en espagnol

Il existe aujourd'hui une liste non officielle de PostgreSQL en espagnol. Pour s'y inscrire, l'utilisateur doit envoyer un message à :

majordomo@tlali.iztacala.unam.mx

Le corps du message contiendra une seule ligne :

inscripcion pgsql-ayuda

© copyright Éditions Eyrolles

Outils graphiques pour PostgreSQL

Quand PostgreSQL est installé, on a souvent envie d'installer d'autres paquetages ou des outils d'interface graphiques pour commencer à se servir du système de base de données. Le navigateur web est appelé à être le terminal graphique le plus répandu à l'avenir. Le code sera probablement écrit sur le serveur web dans un langage de script compilé tel que le compilateur PHP+Zend, en HTML, en DHTML et avec un peu de JavaScript et d'applets Java du côté du client web. Dans cette optique, il est conseillé de migrer ses applications existantes sous Windows 95/NT vers le compilateur PHP+Zend.

Les meilleurs outils sont, par ordre de préférence :

- Script PHP et compilateur Zend : le compilateur PHP+Zend ;
- X-Designer supporte C++, Java et MFC : http://www.ist.co.uk/xd ;
- Qt pour Windows95 et UNIX : http://www.troll.no et ftp://ftp.troll.no ;
- Code Crusader est disponible sur le CD-ROM Linux, une version freeware de Code Warrior de MetroWorks : http://www.kaze.stetson.edu/cdevel/code_crusader/about.html ;
- Code Warrior de MetroWorks : http://www.metrowerks.com ;
- GNU Prof C++ IDE de (Red Hat) Cygnus : http://www.cygnus.com ;
- Borland C++ Builder pour Linux : http://www.inprise.com ;
- Borland Java JBuilder pour Linux : http://www.inprise.com.

Les choix de langage sont, par ordre de préférence :

- Langage de script PHP Web server, HTML, DHTML avec scripts en Javascript sur le client et les applets Java ;
- Langage de script Perl, avec Perl-Qt ou l'interface de base de données Perl Perl-Tk.

Pour le langage de programmation le plus répandu, C++ (GNU g++) :

- Fast CGI (écrit en GNU C++) avec un frontal web/graphique en Javascript/applets Java ;
- GNU C++ et QtEZ ou QT ;
- GNU C++ avec Lesstiff ou Motif ;
- Le langage Java, mais les programmes Java sont très lents (20 fois plus lents que C++ !) et il faut une licence payante.

D'autres outils sont disponibles : PostgreSQL a une bibliothèque d'interface Tcl/Tk dans la distribution appelée pgTcl. Voir aussi :

- Lesstiff Motif : ftp://ftp.redhat.com/pub/contrib/i386/lesstiff*.rpm ;

© copyright Éditions Eyrolles

- Tcl/Tk : http://www.scriptics.com ;
- Extension orientée objet de Tcl appelée INCR : http://www.tcltk.com ;
- Java FreeBuilder : ftp://ftp.redhat.com/pub/contrib/i386/free*.rpm ;
- SpecTCL : ftp://ftp.redhat.com/pub/contrib/i386/spec*.rpm ;
- Java RAD Tool pour PostgreSQL Kanchenjunga : http://www.man.ac.uk/whaley/kj/kanch.html.

Pilotes ODBC, JDBC et UDBC

Les utilisateurs de PostgreSQL demandent parfois à l'administrateur d'installer des pilotes d'accès à la base de données tels que ODBC, UDBC, etc. ODBC (*Open DataBase Connectivity*), créé par Microsoft, est un standard répandu d'accès aux informations des bases de données des divers éditeurs. Les applications écrites à l'aide des pilotes ODBC sont censées fonctionner sur des bases de données aussi diverses que PostgreSQL, Oracle, Sybase, Informix, etc. Voici les sources disponibles pour se procurer ODBC :

- PostODBC est compris dans la distribution, sur le CD-ROM PostgreSQL. Voir le site principal web : http://www.postgresql.org.
- Open Link Software Corporation (http://www.openlinksw.com) commercialise ODBC pour PostgreSQL et d'autres bases de données, et propose aussi une version gratuite d'ODBC (nombre limité de sessions) — à voir.
- Insight ODBC pour PostgreSQL (http://www.insightdist.com/psqlodbc) : c'est le site officiel pour PostODBC.
- FreeODBC (http://www.ids.net/~bjepson/freeODBC/) : c'est une version gratuite d'ODBC.

UDBC est une version statique d'ODBC, indépendante des gestionnaires de pilotes et du support DLL, qui intègre un support de connectivité aux bases de données directement dans les applications.

- Open Link Software Corporation (http://www.openlinksw.com) commercialise UDBC pour PostgreSQL et d'autres bases de données, et propose aussi une version gratuite d'UDBC (nombre limité de sessions) — à voir.

Java est un langage de programmation indépendant de la plate-forme, développé par Sun Microsystems. Les programmeurs Java ont intérêt à écrire les applications de bases de données à l'aide de JDBC (*Java Database Connectivity*) pour faciliter la portabilité sur les bases de données telles PostgreSQL, Oracle, Informix, etc. Si l'on écrit des applications Java, on trouve les pilotes JDBC pour PostgreSQL sur les sites suivants :

© copyright Éditions Eyrolles

- Le pilote JDBC compris dans la distribution PostgreSQL se trouve dans **postgresql-jdbc*.rpm**.
- http://www.demon.co.uk/finder/postgres/index.html propose le module Sun de connectivité Java à PostgreSQL.
- ftp://ftp.ai.mit.edu/people/rst/rst-jdbc.tar.gz.
- http://www.openlinksw.com : Open Link Software Corporation vend JDBC pour PostgreSQL et d'autres bases de données, et propose aussi une version gratuite de JDBC (nombre limité de sessions) — à voir.
- Site JDBC en Grande-Bretagne : http://www.retep.org.uk/postgres.
- Site de FAQ sur JDBC : http://eagle.eku.edu/tools/jdbc/faq.html

La page d'accueil, le guide et les FAQ de JDBC se trouvent respectivement :

- Accueil JDBC : http://splash.javasoft.com/jdbc
- Guide JDBC : http://www.javasoft.com/products/jdk/1.1/docs/guide/jdbc
- FAQ JDBC : http://javanese.yoyoweb.com/JDBC/FAQ.txt

Voir aussi la section « Tester les interfaces Java », page 383.

Les programmeurs Java peuvent aussi consulter deux sites très utiles pour PostgreSQL :

- ftp://ftp.redhat.com/pub/contrib/i386 : voir postgresql-jdbc-*.rpm.
- http://www.blackdown.org.

Interfaces Perl et DBI

PERL (*Practical Extraction and Report Language*) est disponible sur tous les systèmes d'exploitation et toutes les plates-formes matérielles du monde. On peut utiliser Perl sur Windows 95/NT, Apple Macintosh iMac, toutes les variantes d'UNIX (Solaris, HPUX, AIX, Linux, Irix, SCO, etc.), les unités centrales MVS, OS/2, OS/400, Amdahl UTS et beaucoup d'autres.

Interface Perl 5 pour PostgreSQL

L'interface Perl pour PostgreSQL est incluse dans la distribution de PostgreSQL. Voir le répertoire **src/pgsql_perl5.**

Autres ressources :

- À propos de pgsql_perl5, contacter par e-mail : E.Mergl@bawue.de
- Une autre source : ftp://ftp.kciLink.com/pub/PostgresPerl-1.3.tar.gz

© copyright Éditions Eyrolles

- Page d'accueil de Perl : http://www.perl.com/perl/index.html
- Didacticiel Perl ; chercher le titre Tutorial sur : http://reference.perl.com/
- Les FAQ Perl se trouvent sur :
  http://www.yahoo.com/Computers_and_Internet/Programming_
  Languages/Perl/
- RPM de Perl-Qt, interfaces utilisateur graphiques Perl :
  ftp://ftp.redhat.com/pub/contrib/i386
  chercher : PerlQt-1.06-1.i386.rpm
- Perl-Qt, interfaces utilisateur graphiques Perl :
  http://www.accessone.com/ jql/perlqt.html
- Perl-XForms, interfaces utilisateur graphiques Perl :
  ftp://ftp.redhat.com/pub/contrib/i386
  chercher : Xforms4Perl-0.8.4-1.i386.rpm
- Perl-Tk, interfaces utilisateur graphiques Perl :
  ftp://ftp.redhat.com/pub/contrib/i386
- Kits graphiques Perl :
  http://reference.perl.com/query.cgi ?ui
- Interfaces de base de données Perl :
  http://reference.perl.com/query.cgi ?database
- Traducteur de Perl en « C » :
  http://www.perl.com/CPAN-local/modules/by-module/B/
  chercher : Compiler-a3.tar.gz
- Traducteur de shell Bourne en Perl :
  http://www.perl.com/CPAN/authors/id/MERLYN/sh2perl-0.02.tar.gz
- Les traducteurs awk vers Perl (**a2p**) et sed vers Perl (**s2p**) sont inclus dans la distribution Perl.
- Voir aussi les groupe de news PERL sur comp.lang.perl.*

Perl DBI, de quoi s'agit-il ?

L'interface de base de données Perl (DBI, *Database Interface*) est une API (*Application Programming Interface*) destinée à programmer des applications d'accès aux bases de données en langage Perl. Les spécifications de Perl DBI API définissent un ensemble de fonctions, de variables et de conventions qui fournissent une interface de base de données complète et cohérente mais indépendante de la base de données réelle.

Les FAQ de Descartes Hermetica, éditeur de DBI, se trouvent sur
descarte@hermetica.com

© copyright Éditions Eyrolles

DBD-Pg-0.89, pilote DBI pour PostgreSQL

Le fichier du pilote, DBD-Pg-0.89.tar.gz, se trouve sur les sites suivants :

- DBD-Pg-0.89 : http://www.perl.com/CPAN/modules/by-module/DBD/
- Comprehensive Perl Archive Network, CPAN : http://www.perl.com/CPAN
- Liste des pilotes DBI et pages des modules DBI : http://www.hermetica.com/technologia/perl/DBI
- Informations sur DBI : http://www.fugue.com/dbi/
- Site ftp primaire : ftp://ftp.demon.co.uk/pub/perl/db
- Liens divers DBI : http://www-ccs.cs.umass.edu/db.html
- Liens divers DBI : http://www.odmg.org/odmg93/updates_dbarry.html
- Liens divers DBI : http://www.jcc.com/sql_stnd.html
- Base de données PostgreSQL : http://www.postgresql.org

Support technique pour DBI

Envoyer commentaires et rapports de bogue à E.Mergl@bawue.de. Inclure la sortie de **perl -v** et de **perl -V**, la version de PostgreSQL, la version de DBD-Pg et la version de DBI dans le rapport de bogue.

DBI, Perl et DBperl, de quoi s'agit-il ?

Pour paraphraser Tim Bunce, architecte et auteur de DBI :

« DBI est une interface de programmation d'applications (API) d'accès aux bases de données pour le langage Perl. La spécification DBI API définit un ensemble de fonctions, de variables et de conventions qui fournissent une interface de base de données complète et cohérente mais indépendante de la base de données réelle qui est utilisée ».

Traduction : l'interface DBI permet aux utilisateurs d'accéder aux différents types de base de données de manière transparente. Ainsi, que l'on se connecte à Oracle, Informix, mSQL, Sybase ou n'importe quelle autre base de données, on n'a pas besoin de connaître le mécanisme sous-jacent de la couche L3G : l'API définie par DBI fonctionne sur tous les types de bases de données. Un autre avantage est la capacité de se connecter à deux bases de données d'éditeurs différents avec le même script Perl : par exemple, lire des données d'une base de données Oracle et les insérer dans une base de données Informix, le tout en un seul programme. La couche DBI permet de le faire de manière simple et puissante.

© copyright Éditions Eyrolles

Spécifications DBI

Il existe quelques sources d'information sur DBI :

- Spécification DBI : http://www.hermetica.com/technologia/perl/DBI/doc/dbispec
- Informations sur les listes de diffusion de DBI : http://www.hermetica.com/technologia/perl/DBI/tidbits
- Site web du Perl DBI Journal : http://www.tpj.com
- *The Perl5 Database Interface*, un ouvrage écrit par Alligator Descartes et publié chez O'Reilly and Associates.

Les listes de diffusion auxquelles on peut s'inscrire sont :

- Liste de diffusion : http://www.fugue.com/dbi
- dbi-announce : envoyer un e-mail à dbi-announce-request@fugue.com (corps du message : **subscribe**)
- groupe de discussion général dbi-users : envoyer un e-mail à dbi-users-request@fugue.com (corps du message : **subscribe**)
- Archives des listes de diffusion américaines : http://outside.organic.com/mail-archives/dbi-users/
- Archives des listes de diffusion européennes : http://www.rosat.mpe-garching.mpg.de/mailing-lists/PerlDB-Interest

Pour le support commercial et la formation, s'adresser à :

- http://www.perl.co.uk/tpc

Guides pour PostgreSQL

Un administrateur PostgreSQL doit avoir une solide connaissance des commandes SQL, car il se sert directement de SQL pour manipuler les tables système de PostgreSQL. PostgreSQL est accompagné d'un guide SQL, dont les scripts se trouvent dans le répertoire **src/tutorial**.

Ressources sur l'Internet

Le guide SQL se trouve sur http://w3.one.net/jhoffman/sqltut.htm.

Les commentaires et suggestions peuvent être adressés à Jim Hoffman : jhoffman@one.net.

Il propose les sites suivants :

© copyright Éditions Eyrolles

- Référence SQL :
  http://www.contrib.andrew.cmu.edu/ shadow/sql.html
- Interroger SQL Pro :
  http://www.inquiry.com/techtips/thesqlpro/
- Sites utiles sur les bases relationnelles SQL Pro :
  http://www.inquiry.com/techtips/thesqlpro/usefulsites.html
- Sources destinées aux programmeurs :
  http://infoweb.magi.com/ steve/develop.html
- Sites SGBD :
  http://info.itu.ch/special/wwwfiles
  voir le fichier : comp_db.html
- Composants de base de données :
  http://www.compapp.dcu.ie/databases/f017.html
- Edition sur le Web :
  http://www.stars.com/Tutorial/CGI/
- Dictionnaire informatique :
  http://wfn-shop.princeton.edu/cgi-bin/foldoc
- DBMS Lab/Links :
  http://www-ccs.cs.umass.edu/db.html
- FAQ de SQL :
  http://epoch.CS.Berkeley.EDU:8000/sequoia/dba/montage/FAQ
  voir le fichier : SQL_TOC.html
- Bases de données SQL :
  http://chaos.mur.csu.edu.au/itc125/cgi/sqldb.html
- Page de conception des bases de données RIT :
  http://www.it.rit.edu/wjs/IT/199602/icsa720/icsa720postings.html
- Site de liens vers les bases de données :
  http://www.pcslink.com/ej/dbweb.html
- Guides de programmation sur le web :
  http://www.eng.uc.edu/jtilley/tutorial.html
- Ressources de développement :
  http://www.ndev.com/ndc2/support/resources.htp
- Liste de requêtes : http://ashok.pair.com/sql.htm
- MAGE SQL divers :
  http://jazz.external.hp.com/training/sqltables/main.html
- Liste de ressources Internet
  http://www.eit.com/web/netservices.html

© copyright Éditions Eyrolles

Sites de référence de PostgreSQL

Le site principal se trouve à : http://www.postgresql.org.

Le HOWTO de la base de données PostgreSQL se trouve à :
http: ://www.metalab.unc.edu/LDP/HOWTO/PostgreSQL-HOWTO.html.

Pour apprendre pourquoi Linux est considéré comme un meilleur serveur de base de données que Windows 95/NT, voir :
http://www.aldev.8m.com ou http://www.aldev.webjump.com.

Résumé

Quand on a exploré toutes les bases de données gratuites, dont le code source est disponible, on s'aperçoit que PostgreSQL est le système de base de données objet relationnel SQL le plus évolué, le plus utilisé et le plus robuste au monde.

PostgreSQL présente l'avantage que le gros du travail est déjà fait. Il dispose de pilotes ODBC et JDBC, dont on peut se servir pour écrire des applications indépendantes des bases de données. Les applications PostgreSQL qui se servent des pilotes ODBC et JDBC sont facilement portables sur d'autres bases de données telle Oracle, Sybase et Informix, et vice versa.

On peut se demander : pourquoi PostgreSQL ? La réponse est évidente : comme il faut beaucoup de temps pour développer un système de base de données à partir de zéro, le bon sens commande de choisir un système de base de données qui satisfait déjà les conditions suivantes :

- le code source est disponible (selon le système de code *open-source*) ;
- aucune chaîne de licence ni de propriété associée ;
- distribué sur l'Internet ;
- en développement depuis plusieurs années ;
- conforme aux standards tels qu'ISO/ANSI SQL 92 (et SQL 89) ;
- conforme aux besoins futurs tels que SQL 3 (SQL 98)
- des fonctions avancées.

PostgreSQL satisfait à toutes ces conditions, c'est donc le logiciel qui répond à la question.

© copyright Éditions Eyrolles

FAQ

Q. Pourquoi, quand je lance **initdb**, « ça plante » ?

R. Vérifiez que vous n'avez pas de binaires d'une version précédente dans votre chemin (si le message **WARN:heap_modifytuple : repl is \9** apparaît). Vérifier que l'ensemble des chemins est correct et que l'utilisateur Postgres possède bien les bons fichiers.

Q. Comment installer PostgreSQL ailleurs que dans **/usr/local/pgsql** ?

R. La méthode la plus simple est de spécifier l'option **-prefix** quand on exécute la configuration. Si l'on a oublié de le faire, éditer le fichier **Makefile.global** et modifier **POSTGRESDIR** en conséquence, ou créer un fichier **Makefile.custom** et y définir **POSTGRESDIR**.

Q. Quand je démarre postmaster, je reçois un message *Bad system call* ou un dump du noyau. Pourquoi ?

R. Cela peut être dû à divers problèmes, mais la première chose à faire est de s'assurer que l'on a bien installé les extensions système V dans le noyau. PostgreSQL a besoin d'un noyau qui supporte la mémoire partagée et les sémaphores.

Q. Quand j'essaie de démarrer postmaster, je reçois un message d'erreur *IpcMemoryCreate.* Pourquoi ?

R. Soit vous n'avez pas configuré correctement la mémoire partagée dans le noyau, soit il vous faut l'agrandir. La taille nécessaire dépend à la fois de l'architecture et du nombre de buffers et de processus de back-end configurés pour postmaster. Dans la plupart des systèmes, avec le nombre de buffers et de processus par défaut, il faut un minimum de 1 Mo environ.

Q. Quand j'essaie de démarrer postmaster, je reçois un message d'erreur *IpcSemaphoreCreate.* Pourquoi ?

R. Si le message d'erreur est *IpcSemaphoreCreate : semget failed (No space left on device)*, c'est que le nombre de sémaphores du noyau est insuffisant. Postgres a besoin d'un sémaphore par processus de back-end. À titre de solution provisoire, démarrer postmaster avec une limite plus petite que le nombre de processus de back-end. Utiliser **-N** avec un paramètre inférieur à la valeur par défaut, qui est de 32. La bonne solution, ensuite, est d'augmenter les paramètres **SEMMNS** et **SEMMNI** du noyau.

S'il s'agit d'un autre message d'erreur, c'est que vous n'avez pas configuré du tout le support des sémaphores dans votre noyau.

© copyright Éditions Eyrolles

Q. Comment puis-je éviter que d'autres hôtes accèdent à ma base de données PostgreSQL ?

R. Par défaut, PostgreSQL n'autorise que les connexions de la machine locale, à l'aide de sockets du domaine UNIX. Les autres machines ne pourront pas se connecter sauf si vous ajoutez le flag **-i** au lancement du postmaster et que vous autorisez l'authentification des noms d'hôte en modifiant le fichier **$PGDATA/pg_hba.conf** en conséquence, pour permettre les connexions TCP/IP.

Q. Pourquoi ne puis-je pas me connecter à ma base de données depuis une autre machine ?

R. La configuration par défaut n'autorise que les connexions de la machine locale à l'aide de sockets UNIX. Pour autoriser les connexions TCP/IP, assurez-vous que le postmaster a été démarré avec l'option **–i** et qu'une entrée a été ajoutée au fichier **pgsql/data/pg_hba.conf** pour l'hôte concerné. Voir le manuel de **pg_hba.conf.**

Q. Pourquoi ne puis-je pas accéder à la base en tant qu'utilisateur root ?

R. Vous ne devez pas créer d'utilisateurs de bases de données avec l'id utilisateur 0 (root), qui ne peuvent pas accéder à la base de données. C'est une règle de sécurité : sinon, tout utilisateur pourrait lier dynamiquement des modules objet dans le moteur de la base de données.

Q. Tous mes serveurs plantent sur l'accès concurrent à une table. Pourquoi ?

R. Ce problème est sans doute dû à un noyau qui n'est pas configuré pour le support des sémaphores.

Q. Comment régler le moteur de base de données pour de meilleures performances ?

R. Les index peuvent certainement accélérer les requêtes. La commande **explain** permet de voir comment PostgreSQL interprète une requête, et quels index sont utilisés.

Si l'on fait des insertions en masse, il vaut mieux les faire dans un batch qui se sert de la commande **copy** : c'est beaucoup plus rapide que les insertions individuelles. D'autre part, les instructions qui ne sont pas placées dans un bloc de transaction **begin work/commit** sont considérées comme des transactions isolées : il vaut donc mieux exécuter plusieurs instructions en un seul bloc de transaction. Cela réduit la charge liée aux transactions. Il faut aussi envisager de supprimer et de recréer les index quand on modifie les données en grand nombre.

On peut effectuer plusieurs réglages : désactiver **fsync()** en démarrant postmaster avec les options **-o –F**, ce qui évite que **fsync()** actualise le disque après chaque transaction.

© copyright Éditions Eyrolles

L'option –**B** de postmaster permet d'augmenter le nombre de buffers de mémoire partagée utilisés par les processus de back-end. Si l'on augmente trop ce paramètre, le postmaster ne peut plus démarrer parce que l'on a dépassé la limite du noyau pour l'espace de mémoire partagée. Par défaut, il y a 64 buffers qui occupent 8 Ko chacun.

L'option –**S** du back-end permet d'augmenter la valeur maximale de mémoire utilisée par chaque processus de backend, pour des tris temporaires. La valeur -**S** se mesure en kilo-octets et, par défaut, elle est de 512 Ko. Il est imprudent de trop augmenter cette valeur, car on peut manquer de mémoire au moment où une requête appelle plusieurs tris simultanément.

La commande **cluster** permet de grouper les données de tables de base pour constituer un nouvel index. Voir le manuel de **cluster(l)** pour plus de détails.

Q. Quelles sont les fonctionnalités de débogage de PostgreSQL ?

R. PostgreSQL propose plusieurs fonctionnalités utiles au débogage.

D'abord, lors de la configuration, l'option **—enable-cassert** active des **assert()** qui surveillent la progression du back-end et l'interrompent en cas d'incident.

Postgres et postmaster ont plusieurs options de débogage. D'abord, chaque fois que l'on démarre le postmaster, on peut envoyer la sortie standard et la sortie d'erreur vers un fichier de trace, à l'aide des commandes :

```
cd /usr/local/pgsql
. /bin/postmaster >server.log 2>&1 &
```

Cette instruction place un fichier **server.log** dans le répertoire principal de PostgreSQL. Ce fichier contient des informations concernant les problèmes ou erreurs rencontrés par le serveur. Postmaster dispose de l'option –**d** pour des informations plus détaillées ; elle comprend un nombre qui spécifie le niveau de débogage. Il faut savoir que les plus hautes valeurs de ce niveau génèrent des fichiers de trace très volumineux.

On peut tester le back-end postgres en l'exécutant depuis la ligne de commande et en entrant directement des instructions SQL ; mais ce n'est pas recommandé, en dehors des phases de débogage. Noter que, dans ce cas, un retour à la ligne termine la requête — et non un point-virgule. Si l'on a compilé avec les options de débogage, on peut utiliser un débogueur et voir ce qu'il se passe. Mais, comme le back-end n'a pas été démarré à partir du postmaster, il ne fonctionne pas dans un environnement identique, et les problèmes d'interaction entre le verrouillage et le back-end ne peuvent pas être reproduits. Certains systèmes d'exploitation peuvent suivre le fonctionnement d'un back-end et procéder directement au diagnostic des problèmes.

© copyright Éditions Eyrolles

Le programme **postgres** propose des options **-s**, **-A** et - **t** pour le débogage et les mesures de performances.

On peut aussi le compiler avec un profil qui repère les fonctions les plus consommatrices de temps machine. Les fichiers de profil de back-end sont placés dans le répertoire **pgsql/data/base/dbname.** Le fichier de profil du client est placé dans le répertoire courant.

Q. J'obtiens le message *Sorry, too many clients* quand j'essaie de me connecter. Pourquoi ?

R. Il faut augmenter le nombre maximum de processus de back-end concurrents à démarrer par postmaster.

Dans Postgres 6.5.*, la limite par défaut est de 32 processus. On peut l'augmenter en redémarrant postmaster avec la valeur voulue dans l'option **-N.** Dans la configuration par défaut, la valeur de **-N** peut atteindre 1024 ; pour disposer d'un nombre supérieur, il faut augmenter la valeur de **MAXBACKENDS** dans **include/config.h** et reconstruire l'exécutable. On peut définir la valeur par défaut de **-N** au moment de la configuration, à l'aide de l'option de configuration **with-maxbackends.**

Si l'on augmente **-N** au-delà de 32, il faut augmenter **-B** au-delà de la valeur par défaut de 64. Pour un grand nombre de processus de back-end, on sera probablement amené à augmenter d'autres paramètres de configuration de noyau UNIX. Penser à vérifier la taille maximale des blocs de mémoire partagée, **SHMMAX** ; le nombre maximum de sémaphores, **SEMMNS** et **SEMMNI** ; le nombre maximum de processus, **NPROC** ; le nombre maximum de processus par utilisateur, **MAXUPRC** et le nombre maximum de fichiers ouverts, **NFILE** et **NINODE**. On impose à Postgres une limite sur le nombre de processus de back-end pour s'assurer que le système ne manque pas de ressources.

Dans les versions de Postgres antérieures à la version 6.5, le nombre maximum de back-ends était de 64 ; le changer obligeait à reconstruire l'exécutable après avoir modifié la constante **MaxBackendId** dans le fichier **include/storage/sinvaladt.h.**

Q. Que signifient les fichiers **pg_tempNNN.NN**, qui apparaissent dans mon répertoire de base de données ?

R. Ce sont des fichiers temporaires générés par l'exécuteur de requête. Par exemple, s'il faut un tri pour traiter un ORDER BY, et que le tri exige plus d'espace que le paramètre –**S** du back-end ne l'autorise, des fichiers temporaires sont créés pour contenir les données excédentaires.

Les fichiers temporaires seront supprimés automatiquement, sauf si un back-end échoue pendant le tri. Si l'on n'a aucune transaction en cours, il vaut mieux supprimer les fichiers **pg_tempNNN.NN.**

© copyright Éditions Eyrolles

Q. Comment initialiser un **pg_group** ?

R. Actuellement, il n'existe aucun moyen facile d'initialiser les groupes d'utilisateur. On doit explicitement insérer ou mettre à jour un enregistrement dans la table **pg_group.** Par exemple :

```
jolly=> insert into pg_group (groname, grosysid, grolist)
jolly=>     values ('posthackers', '1234', '{5443, 8261}');
   INSERT 548224
jolly=> grant insert on foo to group posthackers;
   CHANGE
jolly=>
```

Les champs de **pg_group** sont :

– **groname** : nom de groupe. Ce nom sera strictement alphanumérique — sans caractère de soulignement ni de ponctuation.

– **grosysid** : id du groupe, de type int4, unique pour chaque groupe.

– **grolist** : liste des id de **pg_user** qui appartiennent au groupe. C'est un champ de type int4[ ].

© copyright Éditions Eyrolles

10

Développer une application web

Dans ce chapitre

- Plates-formes d'application web
- Démarrage
- Enjeux de la conception
- Conception du schéma
- Flux de données
- Connectivité de PHP

© copyright Éditions Eyrolles

Introduction

Ce chapitre est consacré à la construction, du début à la fin, d'une application web, qui ressemble à celle qui est associée aux achats par cartes de crédit dans les magasins (représentée à la figure 10.1). Pour des raisons de taille, cette application ne traitera pas entièrement des problèmes de sécurité, mais nous débattrons quand même des questions inhérentes à une telle application — les versions ultérieures auront une sécurité renforcée. L'application complète — qui comprend le code source, le script de la base de données et les instructions d'installation — est disponible sous la licence publique GNU version 2 (GPLv2), sur les sites www.syngress.com et www.sourceforge.net.

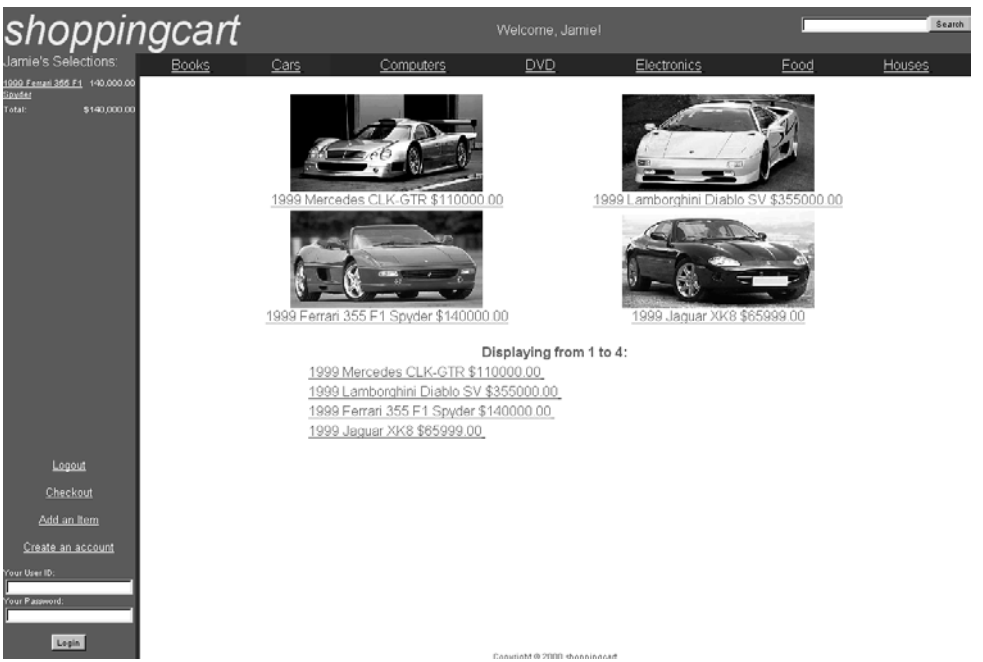

Figure 10–1
Une application d'e-commerce

Plates-formes d'application web

Les plates-formes d'application web disponibles aujourd'hui sont JSP, PHP, Cold Fusion, Zope, ASP, et d'autres encore. Nous allons voir les rapports qu'elles entretiennent, ainsi que leur relation à Linux.

© copyright Éditions Eyrolles

Active Server Pages

Active Server Pages (ASP) de Microsoft est conçu pour Windows NT/2000 mais peut aussi fonctionner (mieux, en fait) sur beaucoup de plates-formes UNIX/Linux, dont Red Hat Linux, *via* l'excellent émulateur de Chili!Soft. ASP peut utiliser tous les objets Component Object Model (COM), mais il s'agit d'une technologie Microsoft. Visual Studio et Internet Information Server de Microsoft représentent un environnement de développement excellent, mais il est plus que perturbé par les problèmes de fiabilité et de sécurité, et n'est finalement pas compatible avec les serveurs web autres qu'IIS, sauf à utiliser Chili!ASP. Pour plus d'informations, visiter le site web de Chili!Soft à http://www.chilisoft.com.

Cold Fusion

Cold Fusion de Allaire est un autre produit propriétaire, facile à apprendre et à utiliser. Bien qu'il n'ait, traditionnellement, été disponible que pour Windows NT, Allaire a récemment commencé sa conversion vers diverses variantes d'UNIX dont HP-UX, Solaris et Linux. Cold Fusion se sert de balises intégrées de type HTML, mélangées au HTML standard, pour bâtir une page personnalisée pilotée par une base de données. Pour plus d'informations, voir le site http://www.allaire.com.

Java Server Pages

Java Server Pages, de Sun, est un nouveau venu dans l'arène des plates-formes d'application web, mais il a le mérite de mettre à disposition la puissance du langage Java de Sun. Bien qu'il soit plein de caractéristiques complexes, c'est un excellent choix si votre entreprise a opté pour les standards Java et CORBA. JSP est disponible sur toute plate-forme qui dispose d'un interpréteur Java, c'est-à-dire Linux, UNIX et Windows NT. Voir http://www.sun.com pour plus d'informations.

Zope

Zope est une excellente plate-forme d'application web *open-source* qui incorpore une base de données hiérarchique, orientée objet, qui comprend des autorisations intégrées et un environnement de développement web. Zope dispose aussi de modules qui se connectent aux plus grands moteurs de bases de données, et offre tous les avantages des bases de données relationnelles traditionnelles orientées objet, ainsi que beaucoup des fonctionnalités de PHP.

© copyright Éditions Eyrolles

De plus, il propose un modèle de contrôle d'exécution pour des applications en aval, ainsi qu'une stratégie d'autorisations intégrée. Zope, comme la plupart des projets *open-source*, se développe très rapidement, et est en train de faire ses preuves. Du point de vue de la complexité et de l'exhaustivité d'un modèle incluant la sécurité, Zope est moins rapide que PHP. Voir le site http://www.zope.org.

Langages de script

De nombreux langages de script *open-source*, tels Perl, Python et TCL, offrent la capacité d'écrire des scripts. Beaucoup d'entre eux peuvent être intégrés dans Apache ou d'autres serveurs web, et éviter les problèmes de sécurité et de performance qui se produisent avec CGI. Ces langages sont un excellent choix pour un développement web, en particulier si on les manie bien. Pour plus d'informations, visiter l'un des sites : http://www.perl.com, http://www.python.org ou http://www.tcl.com.

PHP

PHP est un langage d'application *open-source* extrêmement répandu, destiné à concurrencer directement ASP et JSP. Il propose des modules pour IMAP, POP3, LDAP, XML, PDF, Postscript, FTP, WebDAV, Oracle, MySQL, mSQL, Informix, PostgreSQL, Sybase et beaucoup d'autres moteurs de bases de données, trop nombreux pour tous les citer ici. Il fonctionne sous Linux, UNIX et Windows NT. Du fait que PHP est disponible gratuitement, avec son code source, et qu'il est inclus dans Red Hat Linux, nous développerons l'application en PHP avec l'excellent moteur de base de données MySQL. D'après Netcraft, PHP est implanté sur plus d'un million de serveurs web et progresse très vite. Voir à http://www.php.net pour d'autres informations.

Apache

Apache est, de loin, la plate-forme de serveur web la plus répandue (http://www.apache.org), avec approximativement 60 % de parts de marché d'après Netcraft. IIS de Microsoft vient en seconde position avec 22 % de parts de marché. Apache affiche de grandes performances, mais ce n'est pas son seul atout : il y a une abondance de fonctions additionnelles et de modules disponibles, il est extrêmement stable et offre une bonne sécurité, fonctionne sur tous les systèmes d'exploitation courants, est très extensible, gratuit et *open-source*. Apache s'appuie sur le serveur web NCSA. Les développeurs

© copyright Éditions Eyrolles

d'Apache, peu satisfaits de la lenteur du développement du serveur web NCSA, ont réalisé leur propre version, le Patchy Web Server. Depuis quelques années, Apache développe une plate-forme portable et robuste. La correction des bogues qui affectent la sécurité prend peu de temps, de l'ordre de quelques heures : tout un chacun dispose du code source et peut participer au dépannage, en retour à l'information sur la panne.

Perl et PHP, comme la plupart des autres plates-formes langages *open-source*, sont disponibles en tant que modules plug-in d'Apache. Bien que PHP puisse fonctionner sur d'autres serveurs web, nous allons l'utiliser comme plug-in d'Apache, parce d'autres contraintes de sécurité sont mises en jeu quand on s'en sert *via* CGI.

Démarrage

Consultez, dans votre manuel Red Hat, les instructions d'installation de PHP, de MySQL et des paquetages RPM d'Apache. Dans les versions les plus récentes de Red Hat, ces RPM sont inclus dans les CD-ROM et n'ont pas besoin d'être téléchargés.

Nous recommandons de télécharger et d'installer phpMyAdmin (figure 10.2), qui facilite le développement d'un nouveau système de base de données. PhpMyAdmin est un gestionnaire de base de données de type web pour les bases de données MySQL et propose, sérieux avantage, un code source complet à utiliser quand on développe ses propres applications en PHP. On peut aussi télécharger et installer l'application réelle, disponible gratuitement sur www.syngress.com ou www.sourceforge.net.

Enjeux de la conception

Nous avions certaines idées en tête en concevant notre application d'achat à distance : maintenance facile, extensibilité, clarté du code et une interface conviviale.

Clarté

Comme la clarté est un point sensible, nous avons décidé de ne pas inclure de système sophistiqué de génération d'identifiant de session. Nous avons aussi développé toute l'application sous forme d'un seul script PHP : tout y est accessible, de sorte que l'URL de base du navigateur ne change que s'il se déplace dans le site. Pour distinguer un utilisateur d'un autre, nous créons un

© copyright Éditions Eyrolles

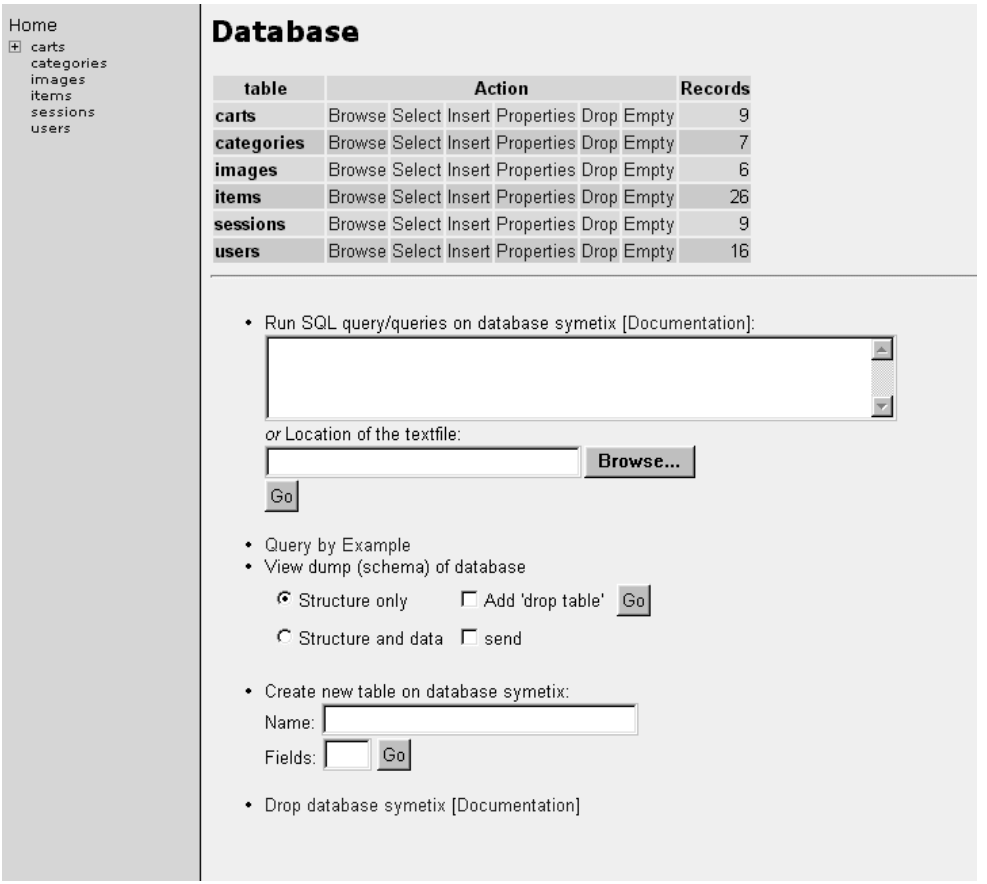

Figure 10–2
Utiliser phpMyAdmin pour développer un nouveau système de base de données

seul cookie par ordinateur : il contient un identifiant de session unique, qui s'incrémente à chaque nouvelle connexion d'un ordinateur, de sorte qu'il n'y a jamais de conflit. Il y a un défaut de sécurité dans ce modèle, dont nous reparlerons à la section relative à la sécurité, plus loin dans ce chapitre.

Extensibilité

L'une des plus grandes barrières qui freinent la maintenabilité est imputable à du code difficile à comprendre. PHP comprend des caractéristiques qui permettent de développer de grands projets, parmi lesquelles de très avancées, tels les objets et les gabarits (dans FastTemplate) que nous n'utilisons pas ici. PHP4 (Zend) et PHP3 (avec PHPLib) proposent aussi des versions plus sûres du point de vue du contrôle de session. Bien sûr, PHP dispose de fonctions

© copyright Éditions Eyrolles

globales ou statiques, de variables locales et d'un certain nombre d'autres caractéristiques conçues pour développer du code modulaire, comme la plupart des langages.

Nous avons développé un noyau avec l'idée d'étoffer le code par la suite, mais beaucoup de travail reste à faire. Vos contributions seront les bienvenues. Par exemple, nous travaillons sur un système de caisse plus élaboré.

Interface conviviale

L'une de nos préoccupations a été d'offrir la capacité de feuilleter les offres du magasin et d'ajouter des éléments à une commande virtuelle sans que l'utilisateur ait à donner d'informations personnelles, tout en lui facilitant certaines tâches : enregistrer, se connecter, choisir des produits dans un ordre quelconque, sans être obligé de retrouver leur référence par la suite. Nous cherchions aussi à simplifier l'administration du magasin.

Conception du schéma

Une version complète, gratuite et sécurisée de cette base de données est en cours de développement *open-source*. Les détails sont disponibles sur www.syngress.com ou www.sourceforge.net. Le code donné ici n'est pas vraiment sécurisé et nous conseillons vivement de ne pas l'implémenter tel quel dans un environnement de production. Mais nous verrons, dans la section qui concerne la sécurité, quelles sont les failles à traiter avant un usage en production.

Nous avons développé une structure de base de données (figure 10.3), conformément aux buts que nous nous sommes fixés. Cette structure de base de données, qui est composée de six tables, propose une définition claire et évidente des données utilisées par la base de données. Reportez-vous à ce schéma lors de la lecture de la section suivante.

Flux de données

Quand l'utilisateur se connecte au site web, une session est créée pour lui et il est doté d'un cookie qui contient une valeur simple, **session_id**. Ce **session_id** contient un numéro unique qui sert de clé primaire dans la table **sessions**. Ici, c'est MySQL qui auto-incrémente la clé primaire ; nous déconseillons cette pratique pour les raisons expliquées dans la section relative à la sécurité.

Quand la session est créée, le champ **session_status** de l'utilisateur a une valeur de 0. Ce **session_status** signifie que l'utilisateur navigue sans être

© copyright Éditions Eyrolles

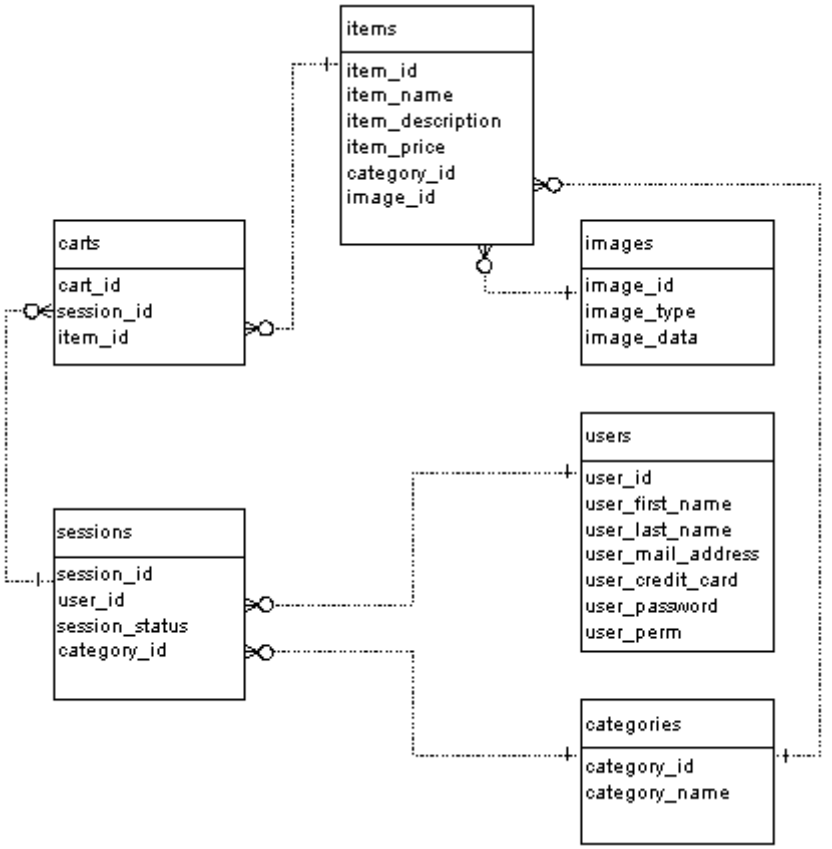

Figure 10–3
Structure de la base de données

authentifié. Comme l'utilisateur doit s'authentifier avant tout achat d'un article, il ne dispose pas encore du bouton de caisse. L'utilisateur est autorisé à feuilleter et ajouter des éléments à sa liste (la table des **carts**) sans authentification. Après authentification (ou enregistrement), **session_status** passe à 1, ce qui signifie que l'utilisateur dispose maintenant du bouton de caisse, et aussi que l'**user_id** est placé dans la table **sessions**.

Quand l'utilisateur clique sur la caisse, **session_status** passe à 2, ce qui signifie que l'utilisateur a achevé la transaction. À ce stade, il est possible d'envoyer des e-mails automatiquement à l'acheteur et au vendeur, indiquant qu'une transaction a eu lieu. Pour des raisons de sécurité évidentes, il ne faut jamais envoyer par e-mail un numéro de carte de crédit non crypté. Le meilleur moyen d'envoyer un e-mail sera une URL unique (par exemple, http://www.shoppingcart.org/view_session_id=323), qui enverra l'image de

© copyright Éditions Eyrolles

la liste dans une page, de telle sorte que personne ne puisse intercepter ce courrier pour obtenir une information sensible, ou à un destinataire protégé par un login. Pour simplifier, nous n'avons pas implémenté ce code dans l'exemple.

PHP, MySQL et Apache : installer l'application

Cette application fonctionnera en PHP compilé, soit comme un module pour Apache (**mod_php**), soit comme un module CGI. Pour vérifier que PHP a été installé correctement sur le serveur web, entrer le texte suivant et l'enregistrer dans un fichier **test.php3** du répertoire Documents du navigateur web :

```
< ?php
phpinfo();
 ?>
```

Si l'on va à http://webserver/test.php3, on voit une grande page qui donne toutes sortes d'informations à propos de l'installation PHP. Dans la section des extensions, vérifier qu'une extension **mysql** est bien chargée. Si ce n'est pas le cas, compiler l'extension **mysql** (voir la documentation MySQL).

Pour créer le schéma de la base de données, créer et charger le script de la base de données. Toutes les tables de l'application sont situées dans la même base. Si elle n'est pas encore créée, choisir un nom et entrer la commande suivante :

```
servername : $ mysql –u username –p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or
Your MySQL connection id is 106270 to server version: 3.22.27
Type 'help' for help.mysql> CREATE DATABASE shopcart
```

Modifier le code des annexes A et B, d'après les instructions qui se succèdent dans les paragraphes qui suivent.

Importer le fichier de script dans MySQL à l'aide de la commande :

```
servername : $ mysql –u username –p shopcart <shopcart.sql
```

Enfin, éditer la fonction **init_db_variables()** dans le fichier **index.php3 (**voir l'annexe A) et indiquer : le nom du serveur de base de données, le nom d'utilisateur, le mot de passe et le nom de la base de données.

Aller à http://webserver/ et s'assurer que les écrans de l'application s'affichent sans erreur. Si l'on obtient une liste de répertoires ou un message

© copyright Éditions Eyrolles

« Permission denied », il faut sans doute ajouter **index.php3** dans la liste DirectoryIndex du fichier httpd.conf d'Apache.

On peut également créer une page HTTP de rafraîchissement, nommée **index.html**, qui transfère immédiatement le navigateur vers **index.php3**.

Connectivité de PHP : faire la connexion

Dans la fonction **connect_db()**, on se connecte au serveur de base de données et on choisit la base de données. Pour faciliter le portage de cette application sur un autre serveur de base de données, il existe des variables séparées dans **init_db_variables()**.

```
MYSQL_CONNECT($db_server,$db_username,$db_password) OR DIE("Unable to
connect to database");

@MYSQL_SELECT_DB("symetix") or die("Unable to select database") ;
```

Lire les données de la base

Voir, dans **browse_column()**, un exemple simple de lecture de données :

```
$result=mysql_query("SELECT * FROM items WHERE category_id=$category_id") ;
$num_results=mysql_num_rows($result) ;
```

Dans la première instruction, on exécute une instruction SQL **SELECT** simple sur la table **items.** Les résultats sont rangés dans la variable **$result**. Dans la seconde instruction, on compte le nombre d'articles reçus.

```
$row=mysql_fetch_array($result)) ;
```

Dans cette instruction, on extrait le premier enregistrement de l'ensemble résultat. On peut alors extraire tous les champs de cet enregistrement en se servant de la commande :

```
$item_id=$row["item_id"] ;
```

Dans ce cas, on a simplement mis à jour la variable **$item_id** par la valeur de l'information contenue dans la ligne courante de **$results**, dans le champ nommé **item_id**. Il y a bien d'autres moyens d'extraire des données d'une chaîne de résultat. PHP est faiblement typé, ce qui signifie qu'il n'y a que peu de distinction entre les variables qui contiennent les divers types d'informations, tels les nombres ou les textes ; PHP se prononce d'après le contexte de la variable, ce qui le rend très facile à coder. Bien sûr, si on le souhaite, on peut toujours forcer le type d'une variable à l'aide de la fonction **settype()**.

© copyright Éditions Eyrolles

Sécurité

N'importe quel pirate remarquera que si son numéro de session est 444, il suffit, dans son cookie personnel, de changer ce numéro en 443 pour détourner la session d'un autre utilisateur. La plupart des applications sécurisées d'achat, ou autres, sur le web combattent ce problème en générant un identifiant unique de session difficile à deviner, en se servant d'une combinaison de la date, de l'heure, d'un nombre aléatoire, de l'adresse IP de l'utilisateur ou de son nom d'hôte, etc. Un schéma compliqué et sophistiqué est théoriquement inviolable.

Quels sont les avantages de Linux par rapport à Windows NT/2000 pour les services web ?

Linux présente tous les avantages traditionnels d'UNIX tels que la stabilité, la capacité et la fiabilité, mais il offre en outre une facilité d'emploi, un grand nombre d'utilitaires puissants et équipés, beaucoup de logiciels gratuits bien utiles, et, à la base, une architecture qui le distingue de NT/2000. Dans Linux, comme dans UNIX, l'accent est mis sur la construction de petits programmes modulaires qui peuvent être reliés rapidement depuis la ligne de commande. Bien que Microsoft ait tenté d'aller dans ce sens depuis de nombreuses années (par exemple, avec les technologies COM, DCOM et DNA), la plupart de ses applications et outils consistent encore en de grandes applications monolithiques qui essaient généralement de tout faire par elles-mêmes. En revanche, les outils Linux sont conçus pour fonctionner ensemble. Si l'on ose une métaphore, cela signifie que, si une brique particulière dont on se sert pour bâtir une tour n'est pas de la bonne couleur, on peut récupérer directement cette brique et la remplacer par une autre, sans que toute la structure ne s'effondre.

L'un des avantages de Microsoft sur Linux réside dans la facilité d'emploi des outils de développement. Mais, cette assertion est sujette à débat. Comme les outils Microsoft facilitent le développement rapide d'une application, c'est la conception qui en souffre, notamment en la faisant reposer sur un cycle continu de prototypes. Comme tout administrateur l'attestera, la conception correcte d'une base de données est absolument cruciale et il ne faut pas commencer réellement le codage avant qu'elle ne soit acquise. Les outils de Microsoft tentent d'épargner à l'utilisateur la complexité réelle de la tâche faite à la main. Pourtant, si l'on y regarde de près, c'est une direction que Linux est en train de prendre ; par exemple, KDE, l'un des bureaux (gestionnaires de fenêtres) Linux les plus répandus, est réputé plus facile à utiliser que Windows 98 lui-même.

© copyright Éditions Eyrolles

Finalement, Linux et UNIX bénéficient d'une immense avance sur NT, à la fois en termes d'installations réelles de grands sites et de support de développement. Alors qu'il y a des milliers et des milliers de programmeurs Visual Basic qui sont passés au développement web, Visual Basic n'a jamais été conçu pour gérer la demande de développement web que l'on attendait. Étant donné que les outils de Visual Basic et Visual Studio déchargent le programmeur de beaucoup de contraintes de conception, beaucoup de programmeurs VB n'ont jamais appris à coder correctement des procédures ou des interfaces de base de données. Cela peut être acceptable pour de petites applications : mais les défauts de conception et les limitations deviennent vite plus apparents sitôt que les applications grossissent.

En dehors de Microsoft, très peu de professionnels des systèmes d'informations contestent que Linux est une plate-forme beaucoup plus stable et équipée que NT/2000. Il y a aussi une piste de migration vers d'autres matériels, les plates-formes non Intel, que NT n'offre pas. Par exemple, Linux est disponible sur les plus grandes plates-formes, dont Sparc de Sun, Alpha de Compaq, PowerPC de Apple, PA-RISC de HP et les processeurs annoncés IA-64 de Intel. Si ces machines ne sont pas suffisantes, IBM supporte Linux sur ses plus grands mainframes. En plus de son développement sur les très grands systèmes, Linux a toujours fonctionné sur Palm Pilot de 3Com, et Linus Torvalds développe une version spéciale de Linux pour le prochain processeur Crusoe Transmeta qui promet de révolutionner les ordinateurs portables, de par la durée de vie exceptionnelle de ses batteries et une architecture de processeur qui s'auto-optimise.

La vérité est qu'en choisissant Linux, vous avez fait le pas dans le bon sens. L'*open-source* est appelé à durer, ce que toute l'industrie a reconnu et a accepté comme un fait.

Conclusion

Dans ce chapitre, une application d'e-commerce a été bâtie à l'aide de MySQL et de PHP. Nous restons attentifs à toute implémentation de ce code. L'application complète — avec le code source, le script de la base de données et les instructions d'installation — est disponible gratuitement sous la licence publique GNU version 2 (GPLv2) sur www.syngress.com ou www.sourceforge.net. Si vous ne connaissez pas la GPL (*GNU General Public License*), sa description, ainsi que le texte de la licence, sont disponibles sur http://www.gnu.org/copyleft/gpl.html.

© copyright Éditions Eyrolles

FAQ

Q. Nous n'avons aucune expertise en PHP dans notre entreprise, mais nous avons quelques experts en Perl. Y a-t-il d'autres applications *open-source* d'e-commerce ?

R. Oui. L'une des applications les plus complètes est Minivend, écrit en Perl (http://www.minivend.com). Une autre excellente application en Perl est OpenCart (http://www.cdrom.com, http://www.opencart.com/), développée et utilisée par Walnut Creek CDROM. Pour d'autres exemples, en PHP ou d'autres langages, consulter le site de Freshmeat.Net sur http://www.freshmeat.net.

Q. Quel est le plus rapide des serveurs web ?

R. Cela dépend sur quoi fonctionne votre serveur et, bien sûr, de ce que vous en attendez. En général, pour un serveur statique, les pages web traditionnelles (telles que les pages HTML traditionnelles), contrairement à celles générées par PHP, ASP, ou d'autres langages, le serveur web le plus simple sera le plus rapide. Les serveurs les plus sophistiqués comprennent des mémoires caches et d'autres caractéristiques, liées à la performance, pour améliorer la vitesse des fonctions les plus vitales (comme la génération de pages dynamiques), mais ces caractéristiques elles-mêmes induisent une certaine charge quand elles ne sont pas nécessaires. En général, les fonctions de sécurité ralentissent elles aussi le serveur. Ainsi, si la vitesse est l'objectif principal, il faudra peser soigneusement les diverses options et les comparer à votre besoin. En général, Apache est une très bonne solution car il offre un excellent équilibre des caractéristiques et des performances.

Q. Est-ce que j'ai besoin de permissions particulières pour utiliser l'application d'e-commerce fournie dans ce chapitre ?

R. L'application d'e-commerce a été conçue sur un serveur hébergé par Hurricane Electric à http://www.he.net. Aucun accès root ou autre accès inhabituel n'est nécessaire.

Q. Où puis-je trouver des professionnels PHP ?

R. Si l'on a des difficultés à trouver des professionnels qualifiés en PHP, le mieux est de les chercher où ils se cachent. Les moyens conseillés sont de poster un message sur les offres d'emploi à http://phpbuilder.com ou http://www.schaffner.net/emp/. On peut aussi envoyer un message à une liste de diffusion PHP.

© copyright Éditions Eyrolles

A

Script PHP de l'application d'e-commerce

© copyright Éditions Eyrolles

Application d'e-commerce

de Jamie Becker

Copyright (c) 2000, Jamie Becker, jamiebecker@ring0.com

Ce programme est un logiciel gratuit : il peut être redistribué et/ou modifié selon les termes de la licence publique générale GNU, publiée par la Free Software Foundation ; soit la version 2 de la licence, soit (au choix) toute version ultérieure.

Ce programme est publié dans le but d'être utile, mais SANS AUCUNE GARANTIE, pas même de garantie implicite quant à son adéquation à une quelconque commercialisation, qu'il s'agisse d'une stricte revente, ou bien d'un arrangement, compte tenu d'une utilisation particulière. Voir la licence publique générale GNU pour plus de détails.

On doit obtenir une copie de la licence publique générale GNU en même temps que ce programme ; si ce n'est pas le cas, écrire à

Free Software Foundation, Inc.,
59 Temple Place - Suite 330,
Boston, MA 02111-1307, USA.

ou consulter les sites www.gnu.org ou www.fsf.org

pour obtenir le texte complet de la licence.

```
<?PHP

// Shopping Cart Application
// By Jamie Becker
// Copyright (c) 2000, Jamie Becker, jamiebecker@ring0.com

// This program is free software; you can redistribute it and/or
// modify it under the terms of the GNU General Public License
// as published by the Free Software Foundation; either version 2
// of the License, or (at your option) any later version.

// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
// GNU General Public License for more details.

// You should have received a copy of the GNU General Public License
// along with this program; if not, write to the Free Software
// Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA  02111-1307, USA.
// Please see www.gnu.org or www.fsf.org for the full text of the license.
```

© copyright Éditions Eyrolles

```
// Source Code modified <80 cols. (JamieBecker,2/16/99)

// strongly recommend this function be separated into a separate, non-public
// file and included here.

function init_db_variables()
{
global $db_server;
global $db_username;
global $db_password;
$db_server='localhost';
$db_username='username';
$db_password='password';
$db_name='shopcart';
}

function set_color_theme()
{
// Set color scheme
global $body_background_color;
global $body_font_color;
global $top_header_color;
global $top_header_font_color;
global $top_header_company_font_color;
global $main_index_tab_color;
global $main_index_tab_font_color;
global $index_tab_color;
global $index_tab_font_color;
global $dividing_hline_color;
global $dividing_vline_color;
global $shopping_cart_color;
global $shopping_cart_font_color;
global $browse_color;
global $browse_font_color;
global $shopping_cart_width; // minimum width
global $shopping_cart_height; // minimum height
global $company_name;
global $search_categories; // if $search_categories=='current', only the
// current category will be searched.
global $table_row_rows;
global $table_row_cols;
$table_row_rows=5;
$table_row_cols=30;
$body_background_color='336699';
$body_font_color='000000';
$top_header_color='336699';
$top_header_font_color='EEEEEE';
$top_header_company_font_color='EEEEEE';
```

© copyright Éditions Eyrolles

```
$main_index_tab_color='990000';
$main_index_tab_font_color='EEEEEE';
$index_tab_color='003366';
$index_tab_font_color='EEEEEE';
$dividing_hline_color='990000';
$dividing_vline_color='990000';
$shopping_cart_color='336699';
$shopping_cart_font_color='FFFFFF';
$browse_color='FFFFFF';
$browse_font_color='336699';
$shopping_cart_width='100'; // minimum width
$shopping_cart_height='0'; // minimum height
$company_name='phpshopcart.sourceforge.net';
// if $search_categories=='current', only the current category will be
// searched.
// $search_categories=='current';
}

// Initiate connection to database
// userid and password should be stored in a separate,
// non-public file and be include()'d here.
function connect_db()
{
global $db_server;
global $db_username;
global $db_password;
MYSQL_CONNECT($db_server,$db_username,$db_password)
   OR DIE("Unable to connect to database");
@MYSQL_SELECT_DB($db_name) or die("Unable to select database");
}

// If an image is requested, display the image and exit. Exit is critical!
function display_image()
{
   global $image_id;
   $result=mysql_query("SELECT * FROM images WHERE image_id=$image_id");
   WHILE ($row=mysql_fetch_array($result))
   {
   $image_type=$row["image_type"];
   $image_data=$row["image_data"];
   }
  header("Content-type: $image_type");
  print $image_data;
  exit();
}
```

© copyright Éditions Eyrolles

```
// Set up cookies
function init_cookies()
{
global $session_id;
if (!isset($session_id)):
   // session_status values:
   // 0 - not logged in, just browsing...
   // 1 - logged in
   // 2 - purchase completed.
   $query = "INSERT INTO sessions (session_status) VALUES('0')";
   $result = MYSQL_QUERY($query);
   $session_id=mysql_insert_id();
   setcookie("session_id",$session_id);
   // Warning - note that this is inherently insecure. For example, it would
   // be easy to simply increment your cookie ID on your browser and hijack
   // someone's session. It would be better to generate a pseudorandom number
   // or a unique ID that couldn't be easily guessed. DO NOT implement this on
   // a production machine.
endif;
}


// Register User in Database
function register_user()
{
global $register_form;
global $user_id;
global $user_first_name;
global $user_last_name;
global $user_mail_address;
global $user_credit_card;
global $user_password;
global $session_id;
If ($register_form!=""):
   // Verify that information is accurate and acceptable for input
   // into the database.
   // (Insert your info checking routines here)
   // Insert the information into database.
   $fields="user_first_name, user_last_name, user_mail_address, ";
   $fields=$fields."user_credit_card, user_password";
   $values="'$user_first_name', '$user_last_name', '$user_mail_address', ";
   $values=$values."'$user_credit_card', '$user_password'";
   $query = "INSERT INTO users ($fields) VALUES($values)";
   $result = MYSQL_QUERY($query);
   $user_id=mysql_insert_id();
   $query="UPDATE sessions SET user_id = '$user_id' WHERE session_id = ";
   $query=$query."'$session_id'";
   $result=MYSQL_QUERY($query);
```

© copyright Éditions Eyrolles

```
Endif;
}


// Log out user
function logout_user()
{
global $session_id;
global $session_status;
global $login_success;
global $logout;
If ($logout):
  $query="UPDATE sessions SET user_id = '' WHERE session_id = '$session_id'";
  $result=MYSQL_QUERY($query);
  $query="UPDATE sessions SET session_status = '0' WHERE session_id = '";
  $query=$query."'$session_id'";
  $result=MYSQL_QUERY($query);
  $session_status=0;
  $login_success=0;
Endif;
}


// Log in user
function login_user()
{
global $login;
If ($login):
  global $user_id;
  global $user_password;
  global $session_id;
  global $session_status;
  global $login_success;
  $query="SELECT user_password from users WHERE user_id='$user_id'";
  $result=mysql_query($query);
  $num_results=mysql_num_rows($result);
  If ($num_results!=0):
    $row=mysql_fetch_array($result);
    $test_password=$row["user_password"];
    If ($test_password == $user_password):
     $query = "UPDATE sessions SET user_id = '$user_id' WHERE session_id ='";
      $query = $query."' '$session_id'";
      $result = MYSQL_QUERY($query);
     $query = "UPDATE sessions SET session_status = '1' WHERE session_id ='";
      $query = $query."' '$session_id'";
      $result = MYSQL_QUERY($query);
      $session_status=1;
      $login_success=1;
```

© copyright Éditions Eyrolles

```
    Else:
      $login_success=0;
    Endif;
  Else:
    $login_success=0;
  Endif;
Endif;
}


// See if User has logged in yet and read info if so.
function read_user_info()
{
global $user_id;
global $user_first_name;
global $user_last_name;
global $user_mail_address;
global $user_credit_card;
global $user_password;
global $user_perm;
global $session_status;
global $session_id;
global $uc_user_first_name;
$query="SELECT user_id from sessions WHERE session_id=$session_id";
$result=mysql_query($query);
$num_results=mysql_num_rows($result);
If ($num_results==1):
   $row=mysql_fetch_array($result);
   $user_id=$row["user_id"];
   $result=mysql_query("SELECT * from users WHERE user_id=$user_id");
   $num_results=mysql_num_rows($result);
   If ($num_results==1):
      $row=mysql_fetch_array($result);
      $user_id=$row["user_id"];
      $user_first_name=$row["user_first_name"];
      $uc_user_first_name=ucwords($user_first_name);
      $user_last_name=$row["user_last_name"];
      $user_mail_address=$row["user_mail_address"];
      $user_credit_card=$row["user_credit_cart"];
      $user_password=$row["user_password"];
      $user_perm=$row["user_perm"];
      $session_status=1;
      $query = "UPDATE sessions SET session_status = '1' WHERE session_id =";
      $query = " '$session_id'";
      $result = MYSQL_QUERY($query);
   Endif;
Endif;
}
```

© copyright Éditions Eyrolles

```
// Layout Page
function layout_page()
{
global $company_name;
global $body_background_color;
global $body_font_color;
Print "<HTML>\n";
Print "<HEAD>\n";
Print "<TITLE>\n$company_name</TITLE>\n";
Print "</HEAD>\n";
Print "<BODY MARGINWIDTH=0 MARGINHEIGHT=0 LEFTMARGIN=0
TOPMARGIN=0 ";
Print "TEXT='$body_font_color' VLINK='$body_font_color' ";
Print "BGCOLOR='#$body_background_color' ";
Print "ALINK='$body_font_color' LINK='$body_font_color'>\n";
}

// Select initial category
function init_category()
{
global $category_id;
global $session_id;
// get category id from current session, if it exists.
If ($category_id=="" || $category_id=="0"):
  $query="SELECT category_id from sessions WHERE session_id=$session_id";
  $result=mysql_query($query);
  $row=mysql_fetch_array($result);
  $category_id=$row["category_id"];
Else:
  // this would get executed if category id was set on the URL, for example
  // when changing categories.
  $query = "UPDATE sessions SET category_id = '$category_id' WHERE ";
  $query = $query."session_id = '$session_id'";
  $result = MYSQL_QUERY($query);
Endif;
// If $category_id is still 0 or "", then go ahead and set it to 1.
If ($category_id=="" || $category_id=="0"):
  $category_id='1';
  $query = "UPDATE sessions SET category_id = '$category_id' WHERE ";
  $query = $query."session_id = '$session_id'";
  $result = MYSQL_QUERY($query);
Endif;
}

// Draw Header Table
function header_table()
{
global $top_header_color;
```

© copyright Éditions Eyrolles

```
global $top_header_font_color;
global $top_header_company_font_color;
global $company_name;
global $search;
global $category_id;
global $session_status;
global $uc_user_first_name;
global $PHP_SELF;
Print "<TABLE BORDER=0 WIDTH='100%' CELLSPACING='0' CELLPADDING='0' ";
Print "BGCOLOR='#$top_header_color'>\n";
// Print "<TR><TD>\n<IMG SRC='img/clear.gif'>\n</TD>\n</TR>\n";
Print "<TR><TD ALIGN='LEFT'>";
Print "<FONT FACE='Helvetica, Verdana, Arial' SIZE=+4 COLOR=";
Print "'$top_header_company_font_color'>";
Print "<EM>$company_name</EM></FONT></TD>\n";
If ($session_status == 1):
   Print "<TD COLSPAN=2 ALIGN='CENTER'><FONT FACE=";
   Print "'Helvetica, Verdana, Arial' COLOR='$top_header_font_color'>";
   Print "Welcome, $uc_user_first_name!</FONT></TD>\n";
Endif;
Print "<TD ALIGN='RIGHT'><FONT FACE='Helvetica, Verdana, Arial'";
Print "COLOR='$top_header_font_color' SIZE=-2><FORM
ACTION='$PHP_SELF'";
Print " METHOD=POST>\n";
Print "<INPUT TYPE=text NAME='search' VALUE='$search'>";
Print "<INPUT TYPE=hidden NAME='current_category_id' VALUE='$category_id'>";
Print "<INPUT TYPE=SUBMIT
VALUE='Search'></FORM></FONT></TD></TR>\n";
Print "</TABLE>\n";
}


// Draw dividing line
function divide_lineh()
{
global $dividing_hline_color;
Print "<TABLE WIDTH=100% BORDER=0 CELLSPACING='0' CELLPADDING='0'>\n";
Print "<TR>\n<TD HEIGHT=1 WIDTH='100%' BGCOLOR='#$dividing_hline_color'>\n";
Print "<IMG SRC='img/clear.gif' HEIGHT=1 WIDTH=1></TD>\n</TR>\n";
Print "</TABLE>\n";
}


// Draw dividing line
function divide_colv()
{
global $dividing_vline_color;
Print "<TD HEIGHT='100%' ROWSPAN=1 WIDTH='1' ";
```

© copyright Éditions Eyrolles

```
Print "BGCOLOR='#$dividing_vline_color'>";
Print "<IMG SRC='img/clear.gif' HEIGHT=1 WIDTH=1></TD>\n";
}


// Checkout!!! Another successful sale!
function checkout()
{
global $session_status;
global $session_id;
$session_status=2;
$query = "UPDATE sessions SET session_status = '2' WHERE session_id = ";
$query = $query."'$session_id'";
$result = MYSQL_QUERY($query);
If ($result):
   Print "<H3>Checkout Successful! Thank you!</H3><HR><BR>";
Else:
   Print "<H3>Checkout Failed! Thank you!</H3><HR><BR>";
Endif;
}


// Start table for shopping cart and browse columns
function start_shop_browse_table()
{
global $browse_color;
Print "<TABLE BORDER=0 WIDTH='100%' CELLSPACING='0' CELLPADDING='0'";
Print " BGCOLOR='#$browse_color'>\n<TR>\n";
}


// Start Shopping Cart Column
function start_shop_column()
{
global $shopping_cart_color;
global $shopping_cart_height;
global $shopping_cart_width;
Print "<TD BGCOLOR='#$shopping_cart_color'
HEIGHT='$shopping_cart_height'";
Print " WIDTH='$shopping_cart_width' VALIGN=TOP>";
}

// Delete an item from the cart before display.
function delete_item()
{
global $delete_cart_id;
If ($delete_cart_id!=""):
   // Delete item from the carts database
```

© copyright Éditions Eyrolles

```
    $query = "DELETE FROM carts WHERE cart_id='$delete_cart_id'";
    $result = MYSQL_QUERY($query);
Endif;
}


// Buy (Add) an item to the cart before display.
function buy_item()
{
global $buy_item_id;
global $session_id;
If ($buy_item_id!=""):
    // Insert this buy item into the carts database
    $query = "INSERT INTO carts (session_id,item_id) VALUES('$session_id',";
    $query = $query."'$buy_item_id')";
    $result = MYSQL_QUERY($query);
Endif;
}


// Print Shopping Cart contents
function print_shopping_cart()
{
    global $session_id;
    global $shopping_cart_font_color;
    global $shopping_cart_width;
    global $PHP_SELF;
    global $session_status;
    global $uc_user_first_name;
    global $user_perm;
    global $total;
    global $items_in_cart;
    $query="SELECT items.item_id, items.item_name, items.item_price, ";
    $query=$query."carts.cart_id FROM items, carts WHERE carts.session_id ";
    $query=$query."='$session_id' AND carts.item_id=items.item_id ORDER BY ";
    $query=$query." items.item_name";
    $result=mysql_query($query);
    If (mysql_num_rows($result)>0):
      If ($session_status>0):
            Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR=";
            Print "'$shopping_cart_font_color'>$uc_user_first_name's ";
          Print "Selections:<BR></FONT>\n";
      Else:
            Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR=";
          Print "'$shopping_cart_font_color'>My Selections:<BR></FONT>\n";
      Endif;
    Endif;
    // start microtable
```

© copyright Éditions Eyrolles

```
    Print "<TABLE WIDTH=100% BORDER=0 CELLSPACING='0' CELLPADDING='0'>\n";
    Print "<TR><TD COLSPAN=2><IMG SRC='img/clear.gif' HEIGHT='10' WIDTH=";
     Print "'$shopping_cart_width'></TD></TR>";
     $items_in_cart=0;
     WHILE ($row=mysql_fetch_array($result))
        {
        $items_in_cart++;
        $cart_id=$row["cart_id"];
        $item_id=$row["item_id"];
        $item_name=$row["item_name"];
        $item_price=$row["item_price"];
        $total=$item_price+$total;
        // allow the opportunity to review an item in the cart.
        Print "<TR><TD VALIGN=TOP>";
        Print "<a href='$PHP_SELF?review_cart_id=$cart_id";
        Print "&reviewbuy_item_id=$item_id'>";
        Print "<FONT FACE='Helvetica, Verdana, Arial' SIZE=-3 COLOR=";
        Print "'$shopping_cart_font_color'>$item_name</FONT>";
        Print "</a></TD>";
        Print "<TD VALIGN=TOP ALIGN=RIGHT><FONT FACE='Helvetica, Verdana,";
        Print "Arial' SIZE=-2 COLOR='#$shopping_cart_font_color'>";
        $item_price_str=number_format($item_price,2,'.',',');
        Print "$item_price_str</FONT></TD>";
        Print "<TR><TD COLSPAN=2><IMG SRC='img/clear.gif' HEIGHT='5' ";
        Print "WIDTH='$shopping_cart_width'></TD></TR>";
        }
     If ($items_in_cart>0):
        Print "<TR><TD VALIGN=TOP>";
        Print "<FONT FACE='Helvetica, Verdana, Arial' SIZE=-3 COLOR=";
        Print "'$shopping_cart_font_color'>Total:</FONT>";
        Print "</TD>";
        Print "<TD VALIGN=TOP ALIGN=RIGHT><FONT FACE='Helvetica, Verdana, ";
        Print "Arial' SIZE=-2 COLOR='#$shopping_cart_font_color'>";
        $total_str=number_format($total,2,'.',',');
        Print "$$total_str</FONT></TD>";
       Print "<TR><TD COLSPAN=2><IMG SRC='img/clear.gif' HEIGHT='5' WIDTH=";
        Print "'$shopping_cart_width'></TD></TR>";
     Endif;
     Print "</TABLE>\n";
     // end microtable
}

// Start Browse Column
function start_browse_column()
{
global $browse_color;
global $browse_font_color;
Print "<TD WIDTH='100%' VALIGN=TOP BGCOLOR='#$browse_color' ROWSPAN=1>";
```

© copyright Éditions Eyrolles

```
Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR='#$browse_font_color'>";
}

// Draw Category Table
function category_table()
{
global $index_tab_color;
global $index_tab_font_color;
global $main_index_tab_color;
global $main_index_tab_font_color;
global $PHP_SELF;
global $category_id;
Print "<TABLE WIDTH=100% BORDER=0 CELLSPACING='0' CELLPADDING='3'>\n";
Print "<TR BGCOLOR='#$index_tab_color'>\n";
$result=mysql_query("SELECT * FROM categories ORDER BY category_name");
WHILE ($row=mysql_fetch_array($result))
{
$col_category_id=$row["category_id"];
$col_category_name=$row["category_name"];
If ($col_category_id==$category_id):
   $category_name=$col_category_name;
   Print "<TD BGCOLOR='#$main_index_tab_color'>\n<CENTER>\n";
   Print "<a href='$PHP_SELF?category_id=$col_category_id'>\n";
   Print "<FONT FACE='Helvetica, Verdana, Arial'";
   Print " COLOR='#$main_index_tab_font_color'>\n";
   Print "$category_name</FONT>\n</a>\n</CENTER>\n</TD>\n";
Else:
   Print "<TD BGCOLOR='#$index_tab_color'>\n<CENTER>\n";
   Print "<a href='$PHP_SELF?category_id=$col_category_id'>\n";
   Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR=";
   Print "'#$index_tab_font_color'>\n";
   Print "$col_category_name</FONT>\n</a>\n</CENTER>\n</TD>\n";
Endif;
}
Print "</TR>\n</TABLE>\n";
}


// Inform the user of userid after registration
function inform_user()
{
global $register_form;
global $user_id;
global $user_first_name;
global $user_last_name;
global $user_mail_address;
global $user_credit_card;
global $user_password;
```

© copyright Éditions Eyrolles

```
global $browse_font_color;
global $browse_color;
If (register_form!=""):
Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR='#$browse_font_color'>\n";
Print "<H1>Please write this down!</H1>";
Print "<CENTER><TABLE BORDER=0 WIDTH=75% VALIGN=TOP CELLSPACING='0'";
Print "CELLPADDING='0' BGCOLOR='#$browse_color'>";
Print "<TR><TD><FONT FACE='Helvetica, Verdana, Arial' COLOR=";
Print "'#$browse_font_color'>Your user ID is:</TD>";
Print "<TD><FONT FACE='Helvetica, Verdana, Arial' COLOR=";
Print "'#$browse_font_color'>$user_id</TD></TR>";
Print "<TR><TD><FONT FACE='Helvetica, Verdana, Arial' COLOR=";
Print "'#$browse_font_color'>Your password is:</TD>";
Print "<TD><FONT FACE='Helvetica, Verdana, Arial' COLOR=";
Print "'#$browse_font_color'>$user_password</TD></TR>";
Print "</TABLE>";
Endif;
}


// Inform a user if he was successfully logged in or out.
function inform_auth()
{
global $login;
global $logout;
global $login_success;
global $uc_user_first_name;
If ($login):
  If ($login_success):
     Print "<H3>Login Successful! Welcome, $uc_user_first_name!</H3><HR><BR>";
  Else:
    Print "<H3>Login Failed!</H3>";
    Print "Unauthorized access is prohibited and will be prosecuted. Please";
    Print " <a href='$PHP_SELF?create_account=1'>";
    Print "create an account</a>. If you already have a valid account, ";
    Print "please try again.<BR><HR><BR>";
  Endif;
Endif;
If ($logout):
  // Logout always succeeds unless there is a database problem.
  // It might be useful to remove the cookie from the user's PC.
  Print "<H3>Logout Successful! Feel free to continue browsing.</H3><HR><BR>";
Endif;
}
```

© copyright Éditions Eyrolles

```
// Input Table Row Form
function row_form($field_title,$field_name,$field_type)
{
global $browse_color;
global $browse_font_color;
global $table_row_rows;
global $table_row_cols;
If ($field_type==""):
  $field_type="text";
Endif;
Print "<TR><TD>";
Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR='#$browse_font_color'>";
Print "$field_title";
Print "</FONT></TD><TD>";
Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR='#$browse_font_color'>";
If ($field_type=="textarea"):
  Print "<$field_type ROWS=$table_row_rows COLS=$table_row_cols NAME=";
  Print "'$field_name'></$field_type>";
Else:
  Print "<INPUT TYPE='$field_type' NAME='$field_name'>";
Endif;
Print "</FONT>";
Print "</TD></TR>\n";
}


// Admin the site
function admin()
{
global $PHP_SELF;
global $browse_color;
global $browse_font_color;
global $category_id;
global $table_row_rows;
global $table_row_cols;
$table_row_rows=5;
$table_row_cols=30;
Print "<CENTER><TABLE BORDER=0 WIDTH=75% VALIGN=TOP CELLSPACING='0'";
Print " CELLPADDING='0' BGCOLOR='#$browse_color'>";
Print "<FORM ACTION='$PHP_SELF' ENCTYPE='multipart/form-data' METHOD=POST>";
Print "<INPUT TYPE=hidden NAME='admin_add' VALUE='1'>\n";
Print "<TR><TD COLSPAN=2 ALIGN=CENTER>";
Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR='#$browse_font_color'>";
Print "<H3>Add an Item</H3></FONT></TD></TR>\n";
row_form("Item Name:","item_name","");
row_form("Item Description:","item_description","textarea");
row_form("Item Price:","item_price","text");
// Make Category Pull-down
```

© copyright Éditions Eyrolles

```
$result=mysql_query("SELECT * FROM categories ORDER BY category_name");
Print "<TR><TD>";
Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR='#$browse_font_color'>";
Print "Category:</FONT></TD><TD>";
Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR='#$browse_font_color'>";
Print "<SELECT NAME='add_item_category_id'>";
WHILE ($row=mysql_fetch_array($result))
{
$col_category_id=$row["category_id"];
$col_category_name=$row["category_name"];
Print "<OPTION VALUE=$col_category_id>$col_category_name";
}
Print "</SELECT></TD></TR>\n";
Print "<TR><TD>";
Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR='#$browse_font_color'>";
Print "Select an Image: (optional)</FONT></TD><TD>";
Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR='#$browse_font_color'>";
Print "<INPUT TYPE='hidden' name='MAX_FILE_SIZE' value=500000>";
Print "<input type=file name=image accept='image/*'></FONT></TD></TR>\n";
Print "<TR><TD COLSPAN=2 ALIGN=CENTER>";
Print "<INPUT TYPE=SUBMIT></FONT></TD></TR>\n";
Print "</FORM>";
Print "</TABLE></CENTER>";
}


// Add an item to the site
function admin_add_item()
{
global $category_id;
global $item_name;
global $item_description;
global $item_price;
global $add_item_category_id;
global $image_id;
global $image; // PHP-provided variable (temporary filename)
global $image_name; // PHP-provided variable (original filename)
global $image_size; // PHP-provided variable (in bytes)
global $image_type; // PHP-provided variable (mime-type: image/gif,image/jpeg)
global $PHP_SELF;
If ($image != 'none'):
  $image_data = addslashes(fread(fopen($image,"r"),filesize($image)));
  // insert mime type into images database and get an id #.
  $fields="image_type, image_data";
  $values="'$image_type', '$image_data'";
  $query = "INSERT INTO images ($fields) VALUES($values)";
  $result = MYSQL_QUERY($query);
  $image_id=mysql_insert_id();
```

© copyright Éditions Eyrolles

```
  Print "<img src='$PHP_SELF?image_id=$image_id'><BR>";
Endif;
$fields="item_name, item_description, item_price, category_id, image_id";
$values="'$item_name', '$item_description', '$item_price', ";
$values=$values."'$add_item_category_id', '$image_id'";
$query = "INSERT INTO items ($fields) VALUES($values)";
$result = MYSQL_QUERY($query);
$item_id=mysql_insert_id();
If ($result):
  Print "Item ID#$item_id addition ($item_name) succeeded.";
Else:
  Print "Item ID#$item_id addition ($item_name) failed.";
Endif;
}


// Create_account
function create_account()
{
global $PHP_SELF;
global $browse_color;
global $browse_font_color;
// Start Create Account Form
// We recommend that this form is secured in secure server!!
Print "<CENTER><TABLE BORDER=0 WIDTH=75% VALIGN=TOP CELLSPACING='0' ";
Print "CELLPADDING='0' BGCOLOR='#$browse_color'>";
Print "<FORM ACTION='$PHP_SELF' METHOD=POST>";
Print "<INPUT TYPE=hidden NAME='register_form' VALUE='1'>\n";
row_form("Your First Name:","user_first_name","");
row_form("Your Last Name:","user_last_name","");
row_form("Your Mail Address:","user_mail_address","");
row_form("Your Credit Card:","user_credit_card","");
row_form("Password:","user_password","");
// OR row_form("Password:","user_password","password");
// Recommend you confirm password if you conceal what is typed
Print "<TR><TD COLSPAN=2 ALIGN=CENTER><INPUT
TYPE=SUBMIT></FONT></TD></TR>";
Print "</FORM>";
Print "</TABLE></CENTER>";
}


// Review an individual item
function review_item()
{
global $review_item_id;
global $PHP_SELF;
global $browse_font_color;
```

© copyright Éditions Eyrolles

```
$result=mysql_query("SELECT * FROM items WHERE
item_id=$review_item_id");
WHILE ($row=mysql_fetch_array($result))
   {
   $item_name=$row["item_name"];
   $item_description=$row["item_description"];
   $item_price=$row["item_price"];
   $image_id=$row["image_id"];
   If ($image_id!='0'):
      Print "<img src='$PHP_SELF?image_id=$image_id'><BR>";
   Endif;
   Print "$item_name<BR>\n$item_description<BR>\n<BR>\n\$$item_price<BR><BR>";
   Print "<a href='$PHP_SELF?buy_item_id=$review_item_id'>\n";
   Print "<FONT FACE='Helvetica,Verdana,Arial' COLOR='#$browse_font_color'>";
   Print "Purchase this Item";
   Print "</FONT>";
   Print "</a>\n";
   }
}


// Review the purchase of an individual item
function review_purchase()
{
global $reviewbuy_item_id;
global $review_cart_id;
global $PHP_SELF;
global $browse_font_color;
$result=mysql_query("SELECT * FROM items WHERE
item_id=$reviewbuy_item_id");
WHILE ($row=mysql_fetch_array($result))
   {
   $item_name=$row["item_name"];
   $item_description=$row["item_description"];
   $item_price=$row["item_price"];
   Print "$item_name<BR>\n$item_description<BR><BR>\$$item_price<BR><BR>";
   Print "<a href='$PHP_SELF?delete_cart_id=$review_cart_id'>\n";
   Print "<FONT FACE='Helvetica,Verdana,Arial' COLOR='#$browse_font_color'>";
   Print "Delete this Item from shopping basket.";
   Print "</FONT>";
   Print "</a>\n";
   }
}
```

© copyright Éditions Eyrolles

```
// Start Browse Column, but for search items
function search_column()
{
global $category_id;
global $category_name;
global $search;
global $lower_limit;
global $browse_font_color;
global $PHP_SELF;
global $search_categories;
If ($lower_limit==""):
    $lower_limit=1;
Endif;
$upper_limit=$lower_limit+4;
If ($search_categories=='current'):
  $query="SELECT * FROM items WHERE item_name LIKE '%$search%' OR item_";
  $query=$query."description LIKE '%$search%' AND category_id=$category_id";
  $result=mysql_query($query);
Else:
  $query="SELECT * FROM items WHERE item_name LIKE '%$search%' OR item_";
  $query=$query."description LIKE '%$search%'";
  $result=mysql_query($query);
Endif;
$num_results=mysql_num_rows($result);
$upper_limit=$num_results+$lower_limit-1;
Print "Displaying $category_name containing '$search' from $lower_limit to ";
Print "$upper_limit:<BR>\n<BR>\n";
WHILE ($row=mysql_fetch_array($result))
   {
   $item_id=$row["item_id"];
   $item_name=$row["item_name"];
   $item_description=$row["item_description"];
   $item_price=$row["item_price"];
   Print "<a href='$PHP_SELF?review_item_id=$item_id&current_category_id=";
   Print "$category_id'>\n";
   Print "<FONT FACE='Helvetica,Verdana,Arial' COLOR='#$browse_font_color'>";
   Print "$item_name \$$item_price";
   Print "</FONT>\n";
   Print "</a>\n";
   Print "<BR>\n<BR>\n";
   }
}
```

© copyright Éditions Eyrolles

```
// Display the normal browse window.
function browse_column()
{
global $category_id;
global $category_name;
global $search;
global $lower_limit;
global $browse_font_color;
global $PHP_SELF;
// display photos first...
$query="SELECT * FROM items WHERE category_id=$category_id AND image_id>0";
$result=mysql_query($query);
$num_results=mysql_num_rows($result);
$max_pics=4;
If ($num_results>0):
   Print ("<BR><CENTER><TABLE WIDTH='80%' BORDER=0>");
   If ($num_results > $max_pics):
        $num_results=$max_pics;
   Endif;
   FOR ($x=1;$x<=$num_results;$x++)
    {
    $row=mysql_fetch_array($result);
    $item_id=$row["item_id"];
    $item_name=$row["item_name"];
    $item_description=$row["item_description"];
    $item_price=$row["item_price"];
    $image_id=$row["image_id"];
    If ($x%2 != 0): // if $x is odd
        Print "<TR><TD><CENTER>\n<a href='$PHP_SELF?review_item_id=$item_id'>";
        Print "<img src='$PHP_SELF?image_id=$image_id' BORDER=0 WIDTH=225>";
        Print "<BR><FONT FACE='Helvetica,Verdana,Arial'";
        Print " COLOR='#$browse_font_color'>";
        Print "$item_name \$$item_price</FONT></a>";
        Print "</TD>\n";
    Else:
        Print "<TD><CENTER>\n<a href='$PHP_SELF?review_item_id=$item_id'>";
        Print "<img src='$PHP_SELF?image_id=$image_id' BORDER=0 WIDTH=225>";
        Print "<BR>";
        Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR=";
        Print "'#$browse_font_color'>";
        Print "$item_name \$$item_price</FONT></a>";
        Print "</TD></TR>\n";
    Endif;
    }
    Print ("</TABLE></CENTER><BR>");
Endif;
If ($lower_limit==""):
   $lower_limit=1;
```

© copyright Éditions Eyrolles

```
Endif;
$upper_limit=$lower_limit+4;
$result=mysql_query("SELECT * FROM items WHERE
category_id=$category_id");
$num_results=mysql_num_rows($result);
$upper_limit=$num_results+$lower_limit-1;
Print "<CENTER><TABLE WIDTH=60% BORDER=0>";
Print "<TR><TD ALIGN=CENTER><B>";
Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR='#$browse_font_color'>\n";
Print "Displaying $category_name from $lower_limit to $upper_limit:</TD></TR>";
Print "</FONT>\n";
Print "</B></TD></TR>";
WHILE ($row=mysql_fetch_array($result))
   {
   $item_id=$row["item_id"];
   $item_name=$row["item_name"];
   $item_description=$row["item_description"];
   $item_price=$row["item_price"];
   Print "<TR><TD>";
   Print "<a href='$PHP_SELF?review_item_id=$item_id'>\n";
   Print "<FONT FACE='Helvetica, Verdana, Arial' COLOR=";
   Print "'#$browse_font_color'>";
   Print "$item_name \$$item_price";
   Print "</FONT>\n";
   Print "</a>\n";
   Print "</TD></TR>\n";
   }
Print "</TABLE>";
Print " </TD></TR>\n";
}


// Display bottom of Shopping Cart column
function end_shop_column()
{
global $session_id;
global $shopping_cart_color;
global $shopping_cart_font_color;
global $shopping_cart_width;
global $PHP_SELF;
global $session_status;
global $uc_user_first_name;
global $user_perm;
global $items_in_cart;
Print "<TR><TD BGCOLOR='#$shopping_cart_color' VALIGN=BOTTOM><p>";
If ($session_status == 1):
   Print "<CENTER><A HREF='$PHP_SELF?logout=1'><FONT FACE=";
   Print "'Helvetica,Verdana,Arial' COLOR='$shopping_cart_font_color'";
```

© copyright Éditions Eyrolles

```
    Print " SIZE=-1>Logout</FONT></CENTER><BR>";
Endif;
If ($session_status > 0 and $items_in_cart > 0):
    Print "<CENTER><A HREF='$PHP_SELF?checkout=1'><FONT FACE=";
    Print "'Helvetica, Verdana, Arial' COLOR='$shopping_cart_font_color' ";
    Print "SIZE=-1>Checkout</FONT></CENTER><BR>";
Endif;
If ($user_perm > 0):
    Print "<CENTER><A HREF='$PHP_SELF?admin=1'><FONT FACE=";
    Print "'Helvetica, Verdana, Arial' COLOR='$shopping_cart_font_color'";
    Print " SIZE=-1>Add an Item</FONT></CENTER><BR>";
Endif;
Print "<a href='$PHP_SELF?create_account=1'>\n";
Print "<FONT FACE='Helvetica, Verdana, Arial' SIZE=-1 COLOR=";
Print "'#$shopping_cart_font_color'>";
Print "<CENTER>Create an account</CENTER>";
Print "</FONT></a>";
Print "<BR>";
Print "<FORM ACTION='$PHP_SELF' METHOD=POST>\n";
Print "<FONT FACE='Helvetica, Verdana, Arial' SIZE=-2 COLOR=";
Print "'$shopping_cart_font_color'>Your User ID:<br> ";
Print "<INPUT TYPE=text NAME='user_id'> <br>\n";
Print "Your Password:<br> <INPUT TYPE=password NAME='user_password'>";
Print " <br>\n";
Print "<INPUT TYPE=hidden NAME='login' VALUE='1'><br>\n";
Print "<CENTER><INPUT TYPE=SUBMIT VALUE='Login'>";
Print "</CENTER></FONT></FORM>\n";
Print "</p></TD>";
}


// Display bottom of Browse column
function end_browse_column()
{
global $browse_color;
global $browse_font_color;
global $company_name;
Print "<TD VALIGN=BOTTOM BGCOLOR='#$browse_color'><FONT FACE='Helvetica,";
Print "Verdana, Arial' COLOR='#$browse_font_color' SIZE=-2><CENTER>Copyright ";
Print "&copy 2000 $company_name</CENTER></FONT></TD>";
Print "</TR>";
}

// End Shopping Cart & Browse Table
function end_shop_browse_table()
{
Print "</TABLE>";
}
```

© copyright Éditions Eyrolles

```
// Start of Program
// Note
// NO PROGRAM STATEMENTS SHOULD EXIST OUTSIDE OF A FUNCTION
// BEFORE THIS SECTION!
init_db_variables();
set_color_theme();
connect_db();
If ($image_id != ""):
  display_image();
  exit();
Endif;
init_cookies();
logout_user();
register_user();
login_user();
read_user_info();
layout_page();
init_category();
header_table();
start_shop_browse_table();
start_shop_column();
delete_item();
buy_item();
print_shopping_cart();
divide_colv();
start_browse_column();
divide_lineh();
category_table();
divide_lineh();
inform_auth();
// Change browse window based on action:
If ($checkout!=""):
   checkout();
Endif;
If ($review_item_id!=""):
   review_item();
ElseIf ($admin!=""):
   admin();
ElseIf ($admin_add!=""):
   admin_add_item();
ElseIf($register_form!=""):
   inform_user();
ElseIf ($reviewbuy_item_id!=""):
   review_purchase();
ElseIf ($create_account!=""):
   create_account();
ElseIf ($search!=""):
   search_column();
```

© copyright Éditions Eyrolles

```
Else:
   browse_column();
Endif;
// Display page footer
end_shop_column();
divide_colv();
end_browse_column();
end_shop_browse_table();

?>
```

© copyright Éditions Eyrolles

B

Script MySQL de l'application d'e-commerce

© copyright Éditions Eyrolles

```
# phpMyAdmin MySQL-Dump
# http://phpwizard.net/phpMyAdmin/
#
# Host: localhost Database : shopcart

# --------------------------------------------------------
#
# Table structure for table 'carts'
#

CREATE TABLE carts (
  cart_id mediumint(9) DEFAULT '0' NOT NULL auto_increment,
  session_id tinyint(4) DEFAULT '0' NOT NULL,
  item_id tinyint(4) DEFAULT '0' NOT NULL,
  PRIMARY KEY (cart_id),
  KEY session_id (session_id, item_id)
);


# --------------------------------------------------------
#
# Table structure for table 'categories'
#

CREATE TABLE categories (
  category_id tinyint(4) DEFAULT '0' NOT NULL auto_increment,
  category_name varchar(15) NOT NULL,
  PRIMARY KEY (category_id),
  KEY category_id (category_id),
  UNIQUE category_name (category_name)
);


# --------------------------------------------------------
#
# Table structure for table 'images'
#

CREATE TABLE images (
  image_id mediumint(9) DEFAULT '0' NOT NULL auto_increment,
  image_type varchar(30) NOT NULL,
  image_data longblob,
  PRIMARY KEY (image_id)
);


# --------------------------------------------------------
#
```

© copyright Éditions Eyrolles

```
# Table structure for table 'items'
#

CREATE TABLE items (
  item_id int(11) DEFAULT '0' NOT NULL auto_increment,
  item_name varchar(30),
  item_description tinytext,
  item_price float(10,2) DEFAULT '0.00' NOT NULL,
  category_id tinyint(4) DEFAULT '0' NOT NULL,
  image_id mediumint(9) DEFAULT '0' NOT NULL,
  PRIMARY KEY (item_id),
  KEY image_id (image_id),
  KEY category_id (category_id)
);


# --------------------------------------------------------
#
# Table structure for table 'sessions'
#

CREATE TABLE sessions (
  session_id int(11) DEFAULT '0' NOT NULL auto_increment,
  user_id int(11) DEFAULT '0',
  session_status tinyint(4) DEFAULT '0' NOT NULL,
  category_id tinyint(4),
  PRIMARY KEY (session_id)
);


# --------------------------------------------------------
#
# Table structure for table 'users'
#

CREATE TABLE users (
  user_id int(11) DEFAULT '0' NOT NULL auto_increment,
  user_first_name varchar(15),
  user_last_name varchar(15),
  user_mail_address varchar(100),
  user_credit_card varchar(20),
  user_password varchar(30) NOT NULL,
  user_perm tinyint(4) DEFAULT '0' NOT NULL,
  PRIMARY KEY (user_id)
);
```

© copyright Éditions Eyrolles

Index

@@IDENTITY, 234

A

Active Server Pages, 17, 429
Advanced Oracle Queuing, 114
Advanced Replication, option
    Oracle8i, 102
ALL, vue du dictionnaire des données
    Oracle, 119
Allaire, 429
ALLOW_ABSOLUTE_DBPATHS, 400
ALTER SYSTEM DISCONNECT,
    Oracle, 116
ALTER SYSTEM KILL SESSION,
    Oracle, 116
ALTER USER IDENTIFIED, Oracle, 116
Apache, 17, 430
application web
    conception, 431, 433
    exemple, 428, 439
    installer l'exemple, 435
    plates-formes, 428, 431
    sécurité, 437
    structure de la base de données, 434
Applixware, 19
ASE, Adaptive Server Enterprise
    (Sybase), 202
ASP, Association of Shareware
    Professionals, 318
assistance
    constructeurs, 28, 41
    éditeurs de Linux, 41
    Informix, 196
    Linux Care, 28
    MySQL, 290, 291
    PostgreSQL, 410
    versions commerciales, 13
Athlon, processeur, 12
authentification, 55, 56
    Kerberos, 394, 396

B

bande de sauvegarde, 40
base de données
    conception, 231, 233
    et instance, Oracle, 74
    objet-relationnelle, 364
BLOB, Binary Large OBjects, 355
boîtier de commutation KVM, 405
Borland C++ Builder, 413
Borland Java JBuilder, 413

C

Caldera Systems OpenLinux, 21
cartes d'interface réseau, 40
CAT, vue du dictionnaire des données
    Oracle, 120
CATEXP.SQL, script Oracle, 121
CGI, 17
    une interface vers Progress, 356
CHANGE, commande SQL*Plus, 117
Chili!Soft, émulateur, 429
chkconfig, 108, 371
chunks, Informix, 143
C-ISAM, 130, 132
Cobalt Networks, 26
Code Crusader, 413
Code Warrior, 413
Cold Fusion, 429
colonne Identité, 233, 235, 243

© copyright Éditions Eyrolles

commit à deux phases, 356
compatibilité matérielle, 25
Component Object Model (COM), 429
Comprehensive Perl Archive Network (CPAN), 124
comptes
  MySQL, 297, 302
  Oracle, 114, 116
  Sybase, 223, 226
Conectiva Linux, 21
CONNECT, rôle Oracle, 115
CONTAINS, fonction du L4G Progress, 354
contraintes matérielles, 37, 40
  disques, 38
  exemples, 67
  Informix, 144
  interface réseau, 40
  MySQL, 288
  Oracle, 78, 79
  PostgreSQL, 367, 399
  processeur, 38
  RAID, 39, 40
  RAM, 38
  sauvegarde, 40
  Sybase, 205
Corel Linux, 24
Corel WordPerfect, 19
Cox, Alan, 12
CPAN, site web, 124, 210

D

datadbs01, Informix, 144
DB2, 246, 285
  accéder au logiciel, télécharger, 248
  avantages, 246, 247
  centre de contrôle, 247
    configuration, 263, 265
    fenêtre principale, 266
  configurer un client DB2
    paramètres de base de données requis, 278
    paramètres TCP/IP requis, 274–275
    pour communiquer avec un serveur DB2, 273, 281
  dépannage de la connexion, 280
  documentation, 285
  e-business, 247
  éditions
    Connect Personal, 248
    Personal Developer, 248
    Universal Database Personal, 248
    Workgroup, 254
  installer un client, 265, 272
    créer une instance, 269
    fenêtre d'installation de DB2 268
    fenêtre d'instance, 270
    fenêtre de création des services, 270
    fenêtre de résumé, 271
    personnaliser
      la bibliothèque de produit, 269
      les messages de produit, 269
      un client d'administration, 268
    rapport d'état, 272
  installer un serveur version 6.1, 251, 260
    créer le serveur d'administration, 257
    créer une instance, 255
    fenêtre contextuelle d'avertissement, 260
    fenêtre de création des services, 255
    fenêtre de résumé, 259
    personnaliser
      l'édition Workgroup, 254
      la bibliothèque de produit, 255
      les messages de produit, 254
    propriétés de l'instance, 256
    propriétés du serveur d'administration, 258
    rapport d'état, 260
  instance
    auto-démarrage, 257

© copyright Éditions Eyrolles

créer sur un client, 269
créer sur un serveur, 255
message d'erreur SQL10007N "-1393", 262, 283
préparer l'installation sur Red Hat Linux, 249, 251
version 5.2 et version 6.0, 249
version 6.1, 250
programme d'installation, 253, 282
vérifier l'installation, 261, 263
DB2 Connect Personal, 248
DB2 Personal Developer, 248
DB2 Universal Database Personal, 248
DB2, kit des développeurs de logiciel, 248
db2rh61fix.tgz, correctifs DB2, 250
db2rh61gafix.tgx, correctifs DB2, 250
DBA, Oracle
rôle d'administration, 115
vue du dictionnaire des données, 119
dbaccess, Informix, 194
dbassist, assistant de configuration Oracle8i, 100
DBD-Pg-0.89, pilote DBI pour PostgreSQL, 417
DBI, interface de base de données, 124, 209, 416, 418
dbs_dbs, Informix, 144
dbspaces, Informix, 144, 167
DBSPACETEMP, option Informix, 188
dbstart, script Oracle, 106
Debian GNU/Linux, 21, 22
DEFINE, commande SQL*Plus, 117
Descartes Hermetica, FAQ DBI, 416
DHCP, Dynamic Host Configuration Protocol, 52
DICT, vue du dictionnaire des données Oracle, 120
dictionnaire des données
Oracle, 119
Disk Druid, 47
disques, 38
bruts (raw disks), 141, 204
conseils pour Informix, 144
Sybase, 243
disquette de démarrage, 64, 65
d'urgence, 64
distributions, 12, 13, 21, 24
choisir, 28, 37
comparaison, 23
liste non exhaustive, 21
mise à jour, 27
DNS, Domain Name Service, 53
DOS, émulation, 16
DOSEMU, 16
DSS, Decison Support System. *Voir* systèmes d'aide à la décision
dual boot, 27, 31
DUAL, vue du dictionnaire des données Oracle, 119

E

ecpg
et PostgreSQL, 384
EDIT, commande SQL*Plus, 117
ELF, Executable and Linking Format
format binaire, 15
environnement de test, 27
/etc/services
configuration d'Informix, 159
configuration de Progress, 330
configurer pour DB2, 274
Export, utilitaire Oracle, 120
extents
Oracle, 75
Progress, 335

F

Fast CGI, 413
FAT32
et Linux, 48
fdisk, 47, 206
fichiers de données
Informix, 143
Oracle, 75
Oracle, emplacement, 103
Progress, 335
Sybase, 204
formations, 29
Informix, 198

© copyright Éditions Eyrolles

FreeBSD 2.2-RELEASE et PostgreSQL, 380
FreeODBC, 414
fuseau horaire, 54

G

gcc, outil GNU, 294, 374
genclntsh, 127
ghostscript, 379
glibc, bibliothèques, 85, 249, 308
glibcpatch, correctif, 90, 92
Global Knowledge Inc., 29
gmake, 126, 377
Gnome, 60
GNU C++, 413
GNU Prof C++ IDE, 413
GPL, General Public License, 438
  noyau Linux, 11
GRANT, 115, 226, 300
groupadd, 82
groupe d'utilisateurs
  Informix, 197
  Oracle, 127
GUI, Graphical User Interface, 13
gunzip, utilitaire, 292

H

histoire de Linux, 10
HOST, commande SQL*Plus, 118
HTTP Listener, 121
80510patchset, correctif, 90, 91

I

iBCS, 319, 361
IDN, Informix Developer's Network, 200
IDS 7.30, 133
IDS.2000, 133, 134
IIUG, International Informix Users Group, 197
Illustra, 134
Import, utilitaire Oracle, 120
incompatibilités, 19
INCR, extension de Tcl, 414
INDEX, tablespaces Oracle, 103
indexdbs01, Informix, 144
Informix, 130, 199
  accéder au logiciel, télécharger le fichier, 136
  acquisition et mise en place du logiciel, 135, 138
  besoins d'espace, 145
  chunks, 143
  C-ISAM, 130, 132
  comparaison à Oracle, 131
  contraintes matérielles
    disques bruts (raw disks), 141
    fichiers préformatés (cooked), 142, 143
  création des fichiers de données, 143–144
  datadbs01, 144
  dbs_dbs, 144
  dbspaces, 144, 167
  démarrer le moteur, 160, 166
    redémarrer, 189, 191
  FAQ, 199
  groupe international des utilisateurs Informix (IIUG), 197
  IDS 7.30, 133
  IDS.2000, 133, 134
  indexdbs01, 144
  initialisation du disque et démarrage du serveur, 141, 143
  installation et marquage, 138, 140
    d'un serveur IDS, 140
    du client SDK, 139
  logs logiques, 175, 184
  logsdbs, 144, 173, 175
  mémoire, utilisation, 187
  moteur standard (SE), 132
  oninit, 161, 189
  onmode, 179
  onparams, 169
  onspaces, 169
  onstat, 163, 189, 191
  ontape, 171
  options de configuration de $ONCONFIG, 187, 188
  outils, 191, 196

© copyright Éditions Eyrolles

dbaccess, 194
dbschema, 192
finderr, 192
IDSchunks, rapport, 193
IDSspaces, rapport, 192
onlog, 191
onmonitor, 195
physdbs, 144, 168, 173
répertoires, 145
ressources, 196, 198
formations, 198
groupe de news Usenet, 197
presse, 197
site web, 197
support technique, 196
rootdbs, 144
tempdbs, 184, 186
tempdbs01, 144
Informix Dynamic Server (IDS), 132, 134
$INFORMIXDIR, variable
d'environnement, 131
$INFORMIXDIR/etc/
$ONCONFIG 147–159, 187, 188
$INFORMIXDIR/etc/sqlhosts, 159
INFORMIXSERVER, variable
d'environnement, 147
INFORMIXTEMP, espace, 184
Ingres, 134
INIT.ORA, fichier de paramètres
Oracle, 77, 106
INIT<SID>.ORA, 77
initdb,utilitaire PostgreSQL, 378
Insight ODBC, 414
installation
autres versions de Linux, 66
DB2, version 6.1, 251, 263
décisions préliminaires, 37, 41
Informix, 134, 144
MySQL 3.22, 289, 296
Oracle8, 87, 94
Oracle8i, 94, 105
PostgreSQL, 367, 387
Progress 8.3, 318, 331
Red Hat Linux, 41, 66
Sybase SQL Server 11.0.3, 203, 210
X Window, 60, 62
instance
DB2, 255
Oracle, 74
interface graphique utilisateur, 13, 16
Internet, 17
PostgreSQL et, 365
Internet Information Server, 429
Iomega Jaz, 40
ISAM, Indexed Sequential Access
Method, 130
isql, client Sybase, 209

J

Java, 413, 429
et DB2, 263
et Oracle8i, 72
et PostgreSQL, 383
et Sybase, 210
pour Linux, 86
Java FreeBuilder, 414
Java RAD Tool Kanchenjunga, 414
Java Server Pages, 429
JBOD, Just a Bunch Of Disks, 333
jConnect, Sybase, 210
JDBC
avec Sybase, jConnect, 210
et DB2, 246
pour PostgreSQL, 414
JOIN
opérateurs *=, = et =* de
sybase, 235–237
syntaxe SQL, 236
journaux, 76
Oracle, 104
JRE, Java Runtime Environment
pour DB2, 263, 264
télécharger, 86

K

KDE, 60
Kerberos, 394, 396
kill, commande
avec Progress, 350, 352

© copyright Éditions Eyrolles

kpackage, utilitaire KDE, 136
KVM, boîtier de commutation, 405

L

L4G Progress, 318, 352, 361
$LANG,variable d'environnement, 210
langages de script, 430
$LC_ALL,variable d'environnement, 210
LD_LIBRARY_PATH, variable d'environnement, 84, 378
Lesstiff, 413
libclntsh.so.8.0, 127
libpq++ et libpq, 383
Libranet, 21
LILO, Linux Loader
  configuration, 64
Linux
  architectures multiprocesseurs, 15
  avantages sur Windows 95/NT, 420
  avantages sur Windows NT/2000, 437
  caractéristiques de base, 14, 17
  code source, 12
  distribution, 12
  et matériels anciens, 20
  FAQ, 31, 69
  fonctions réseau, 17
  format ELF, 15
  histoire, 10
  interface graphique utilisateur, 16
  marque commerciale déposée, 11
  mémoire virtuelle 32 bits, 15
  mode noyau complètement protégé 16
  multitâche, 15
  multi-utilisateur, 16
  noyau, 11, 13
  noyau originel, 10
  pré-installation pour Oracle, 79, 87
  quand l'utiliser, 18
  services UNIX standard, 17
  stations de travail, 20
  suites bureautiques, 19
  support des disques bruts, 141
  support des services Internet, 17
  système d'exploitation, 10, 33
Linux Mandrake, 21, 23
LinuxPPC, 21
LIST, commande SQL*Plus, 117
lnxpatch, correctif, 90
logsdbs, Informix, 144
LRU, options Informix, 188

M

make, outil GNU, 294, 374
Mandrake, 23
matériels
  anciens, 20, 25
  Voir contraintes matérielles
MAXUMEM, paramètre du noyau, 327
mdrecoverd, démon, 40
mémoire
  Informix, 187
  Oracle, 74, 75
  Progress, 337
  Sybase, 237
mémoire virtuelle 32 bits, 15
Microsoft SQL Server 6.5, 202
Microsoft Visual Studio, 429
migration, 25, 28
  compatibilité matérielle, 25
  coûts, 30
  depuis un autre OS, 27
  environnement de test, 27
  évaluation, 29
  processus, 28, 30
  recyclage des utilisateurs, 25
Minix, 10
mise à jour, 27
mkbootdisk, utilitaire, 64
mots de passe
  masqués, 56
  MySQL, 297
  système, DB2, 258
  système, Oracle, 105
  système, Sybase, 220
MPSQL Motif-Worksheet, 384
multiprocesseurs, 15, 38
  symétriques, 60
multitâche, 15
multi-utilisateur, 16
MySQL, 288, 315, 430

© copyright Éditions Eyrolles

accéder au logiciel, 290
compte mysql, 299
comptes utilisateurs, 297, 302
    privilèges, 298, 301
configuration, 296, 308
contrôle d'accès, 302
dépannage, points à vérifier, 311
distributions
    Alpha, 292
    Bêta, 292
    binaires, 288
    conventions
        d'appellation, 291–292
    Gamma, 292
    RPM, 288
    sources, 288
environnement d'exécution, 309
FAQ, 314
installation d'une
    distribution, 289–296
    binaire, 293
    choix de la version, 289
    RPM, 294
    source, 294, 296
installation du système, 303, 304
licences et assistance, 290, 291
outils, 310
    mysqlaccess, 310
    mysqladmin, 309, 311
    mysqld, 311
    mysqldump, 311
    mysqlimport, 311
    safe_mysqld, 311
performances, 305, 308
    flag –static, 305
    variables de contrôle, 305, 308
problèmes, 308, 312
répertoire, disposition, 296
sécurité, 297, 298
tables système
    db, bases de données, 301
    host, hôtes autorisés, 301
    user, utilisateurs, 300
transactions, 315
    versions
        désapprouvée, 3.21.X, 289
        développement, 3.23.X, 289
        stable, 3.22.X, 289
mysqladmin, outil d'administration de MySQL, 305, 309, 311
mysqld, outil MySQL, 314
mysqldump, outil MySQL, 315

N

NBUFS, paramètre du noyau, 327
Net8, assistant de configuration réseau Oracle8i, 109
netasst, application Java, 109
NetBSD, et PostgreSQL, 380
netca, application Java, 109
Network Assistant, configuration réseau Oracle8i, 108
NFILES, paramètre du noyau, 327
NIC, Network Interface Card, 40
NIS, Network Information System, 56
niveau d'exécution, 107
NLS_LANG, variable d'environnement, 84
NOFILES, paramètre du noyau, 327
nom
    résolution de, 53
noms de connexion
    et utilisateurs, 223
noyau Linux, 11, 13
    licence GPL, 11
    mode protégé, 16
    reconfiguration, 59

O

ObjectManager, Orasoft, 124
objet relationnel, 364
ODBC, Open DataBase Connectivity
    pour PostgreSQL, 384, 414
OFA, Optimal Flexible Architecture, 80, 93, 96, 103, 104
OLTP, OnLine Transaction Processing. *Voir* systèmes transactionnels
$ONCONFIG, variable d'environnement Informix, 148

© copyright Éditions Eyrolles

oninit, Informix, 160
options –ivy, 161
onmode, Informix, 179
option –ky, 167
onmonitor, Informix, 195
onparams, Informix, 169, 170, 172, 175, 181
onspaces, Informix, 169, 185
onstat, Informix, 163, 171
ontape, Informix, 171, 176
Open Client Library de Sybase, 209
Open Client Library pour C, 209
Open Client Library pour Perl, 209
Open Link Software Corporation, 414
OpenLinux, 22
open-source, 10, 200, 290, 364, 365, 429, 430
/opt/data, répertoire Informix, 145
/opt/data/rootdbs, 160
/opt/informix/ids7.30, répertoire Informix, 145
/opt/sybase, répertoire, 207
ORA_NLS33, variable d'environnement, 84
Orac, 124
Oracle, 72, 126
accéder au logiciel, 87
base de données et instance, 73
cache de journalisation, 75
comptes et groupes système, 82, 83
groupe dba, 82
groupe oinstall, 83
groupe OSDBA, 82
comptes SYS et SYSTEM, 105
concepts, 73, 78
configuration
de l'espace d'échange, 81
du fichier de paramètres, 106
du réseau, 108, 109
du système de fichiers, 80, 81
contraintes matérielles, 78, 79
correctifs, 87
créer un compte, 114, 116
dbstart, 106
démarrage et arrêt automatiques, 106–108
dictionnaire des données, 119
et Perl, 124
extents, 75
FAQ, 126
fichiers
de contrôle, 76
de données, 75
de paramètres, 77
identifiants de version, 77
INIT.ORA, fichier de paramètres, 77, 106
INIT<SID>.ORA, 77
installation, 78, 109
Oracle8, 87, 94
Oracle8i, 94, 105
journaux, 76, 104
mémoire, organisation, 74, 75
options d'environnement, 83, 85
$ORACLE_BASE, 77
$ORACLE_HOME, 77, 84
paquetages de compatibilité, 85
paramètres du noyau, 82
PL/SQL, 78
pool partagé, 74
post-installation, 105, 109
pré-installation du système Linux, 79–87
privilèges système, 115
processus, 113, 114
archivage (ARCn), 114
checkpoint (CKPT), 113
écriture de la base de données (DBW0), 113
écriture du journal (LGWR), 113
file des travaux (SNPn), 114
moniteur de processus (PMON), 113
moniteur de queue (QMNn), 114
moniteur système (SMON), 113
récupération (RECO), 114
verrouillage (LCK0), 114
rôles, 115

© copyright Éditions Eyrolles

schéma, 78
segments, 76
de rollback, 76
SID, 77
SQL*Plus, 116, 119
système hybride, 73, 100
tablespaces, 75
utilisation, 110, 121
utilitaires
Export, 120
Import, 120
SQL*Loader, 121
WebDB, 121
zone globale
des programmes (PGA), 75
système (SGA), 74
Voir aussi Oracle8, Oracle8i
Oracle interMedia Text, option, 101
Oracle interMedia, option, 101
Oracle JServer, option, 102
Oracle Names Servers, 109
Oracle Parallel Server, 114
Oracle Server Manager, 110, 112
ALTER DATABASE OPEN, 111
CREATE SESSION, 111
RESTRICTED SESSION, 111
SHOW SGA, 112
SHUTDOWN, 112
SHUTDOWN ABORT, 112
SHUTDOWN IMMEDIATE, 112
SHUTDOWN NORMAL, 112
SHUTDOWN
TRANSACTIONAL, 112
STARTUP, 111
STARTUP MOUNT, 111
STARTUP NOMOUNT, 111
STARTUP OPEN RECOVER, 112
STARTUP PFILE=initSID.ora, 111
STARTUP RESTRICT, 111
Oracle Spatial, option, 102
Oracle Technology Network, 87, 90, 127
Oracle Times Series, option, 101
Oracle Universal Installer, 86
écran d'accueil, 95
lancement du script orainstRootsh, 96
problèmes, 94
Oracle Visual Information Retrieval,
option, 101
$ORACLE_BASE, 77
$ORACLE_HOME, 77, 84
Oracle8
créer les objets de base de données, 93
installation, 87, 94
installer les correctifs, 90, 92
options d'environnement, 84
orainst, programme d'installation, 87
Oracle8i, 72
Advanced Replication, option, 102
dbassist, assistant de
configuration, 100
dialogue d'installation des
privilèges, 98
et Java, 86
installation, 94, 105
installer les correctifs, 98, 99
options d'environnement, 84
orainstRootsh, script, 96
produits complémentaires en
option, 101
scripts de création de base de
données, 99, 104
orainst, programme d'installation
Oracle8, 87
orainstRoot, script Oracle8i, 96
Orasoft, 122, 124
ObjectManager, 124
SQLWork, 123
ORCL, SID d'Oracle, 77
OTN. *Voir* Oracle Technology Network
OUI. *Voir* Oracle Universal Installer

P

Palm Pilot, 24
paramètres du noyau
Oracle, 82
Progress, 324, 328
partitions, 47, 51
ajouter, 48
brute non formatée, 204

© copyright Éditions Eyrolles

conseils, 66
d'échange multiples, 47
modifier, 48
point de montage, 49
sélection d'un lecteur physique, 51
supprimer, 47
Sybase, 206
taille de la partition, 49
type de partition, 50
type Linux natif, 50
PASSWORD, 116
pdksh, paquetage, 249
Penguin Computing, 26
Perl, 430, 439
Practical Extraction and Report Language
et Oracle, 124
et PostgreSQL, 382, 415, 416
et Sybase, sybperl, 209
Perl 5 pour PostgreSQL, 415
Perl-Qt, 413
Perl-Tk, 413
pg_dumpall, utilitaire PostgreSQL, 376
pg_options, options PostgreSQL 389, 391, 408
PGA, Program Global Area, 75
pgaccess, client graphique de PostgreSQL, 372, 381
Phat Linux, 21
PHP, 413, 430, 432, 439
connectivité, 436
PHPLib, 432
phpMyAdmin, 431
physdbs, Informix, 144
pilotes
localisation, 20
PL/SQL, 78
points de montage, 49
pool partagé, 74
Postgres, 134
PostgreSQL, 364, 425
accéder au logiciel, 366
cas de certains systèmes d'exploitation, 380
compte Linux postgres, 371, 375
connexion du client, 407
contraintes matérielles, 367
grandes bases de données, 399
corriger les bogues, 386
démarrer le serveur, 378
emplacement secondaire, 400
et Internet, 365
exemples, 381, 384
fiche d'identité, 364
gérer une base de données
accéder à une base, 402
créer une base, 401
détruire une base, 403
grandes bases, 399, 404
sauvegarde et restauration, 403
installation
de la distribution source 374, 381
détails, 369, 374
rapide, 367, 369
validation, 381, 384
vérification, 385, 387
interfaces Perl et DBI, 415
listes de diffusion, 411, 412
messages de débogage, 408
mise à jour d'un système existant, 376
options de compilation, 377
outils graphiques, 413
paquetages RPM, 369
pg_group, 396
pg_options, 389, 391, 408
pilotes ODBC, JDBC et UDBC, 414
postmaster, 368, 388, 406
répertoire pgsql, 387
répertoires du système, 387
ressources sur l'Internet, 410, 420
s'assurer de la fiabilité, 386
sécurité, 392, 399
authentification de l'utilisateur, 392
authentification Kerberos, 394–396
contrôle d'accès, 393, 397
créer des groupes, 396

© copyright Éditions Eyrolles

créer des utilisateurs, 396
des connexions TCP/IP, 398
fonctions et règles, 398
méthodes d'authentification, 394
sites de référence, 420
suggestions, 379
support technique, 410
tests de régression, 381, 385
tutoriels, 418
postmaster, processus PostgreSQL, 368
PostODBC, 414
PROBKUP, programme de sauvegarde Progress, 344
processeurs 64 bits, 400
processus
Oracle, 113, 114
Progress, 318, 362
base de données
vide, 336
void, 336
besoin de mémoire, 337
commande kill, 350, 352
configuration, 331, 339
calibrage du pool de buffers, 337
créer une base multivolume, 335–337
espace disque et débit d'entrée/sortie, 333, 334
coût de propriété, 354
dictionnaire, 362
éditions, 323, 360
gérer la base de données, 339, 350
arrêter la base, 343
démarrer la base, 340
démarrer une session, 342
sauvegarder la base, 344
surveillance, 345, 350
images avant et après, 333
indexation des mots, 354
installation, 319, 331
conditions préalables, 319
de l'environnement, 328, 331
du mode client-serveur, 330
du terminal PROTERMCAP, 329
écran d'accueil, 320
options d'installation, 321
saisie des numéros de produits et des codes de contrôle, 321
sélection de la langue, 322
sélection des produits, 323
variables d'environnement, 329
internationalisation, 354
L4G, 352
orientation OLTP, 353
paramètres du noyau, 324, 328
mémoire partagée, 325
paramètres divers, 327
sémaphores, 326
restauration après panne, 353
sauvegarde en ligne, 355
structure de répertoire, 331
types de fichiers ou extents, 335
version 8.3, 318
version 9.1, 318, 353, 355
Progress Email Group, PEG, 319, 345
Progress Software Corporation, PSC, 318, 352, 360
progress.cfg, fichier de configuration, 362
PROMON, moniteur Progress, 345, 350
prostrct, utilitaire Progress, 335
protégé, noyau, 16
PROTERMCAP, fichier de configuration de Progress, 329
protocole de transfert de fichier, FTP, 18
PSC, Progress Software Corporation
psql, terminal de PostgreSQL, 372
Python, 430
et PostgreSQL, 382

Q

Qt pour Windows95, 413
QtEZ, 413

R

RAID, 37, 39, 40, 305
coûts, 39
matériel ou logiciel, 39
Progress et, 333, 334, 360
RAM, 38

© copyright Éditions Eyrolles

et partitions d'échange, 47
Red Hat Linux, 19, 21, 22, 36, 70, 430
et Oracle, 85
et Sybase, 203
installation, 41, 66
authentification, 55, 56
d'un serveur,45
d'une station de travail,44
du réseau,51, 54
mise à niveau, 46
option Tout installer, 60
personnalisée, ou manuelle, 46–66
rapide, 41, 46
sélection de la langue, 42
sélection de la souris, 43
sélection du clavier, 43
sélection du fuseau horaire, 54
type d'installation, 44
installer PostgreSQL, 367, 381
paquetages de base, 57
paquetages optionnels, 58
partitions, 47, 51
préparer l'installation de DB2, 249–251
requêtes parallèles
Progress et, 355
réseau, 17
activation au démarrage, 52
configuration, 51, 54
configuration avec Oracle, 108, 109
DHCP, 52
nom d'hôte, 53
RESOURCE, rôle Oracle, 115
REVOKE, 115, 226
rôles, Oracle, 115
ROLLBACK, tablespaces Oracle, 103
root
configuration du compte, 54
et MySQL, 293, 294, 297, 303, 308
et Oracle, 82, 83, 89, 96, 97, 126
et PostgreSQL, 378
rootdbs, Informix, 144
rpm, outil Red Hat Linux, 294
RPM, Red Hat Package Manager, 57, 288
RUN, commande SQL*Plus, 117

S

safe_mysqld, outil MySQL, 311
sauvegarde
matériels, 40
schéma
Oracle, 78
SCO OpenServer, 319
SCSI, lecteurs, 51, 205
SE, Standard Engine (Informix), 132
segments, Oracle, 76
de rollback, 76
SEMMNI, paramètre du noyau, 326
SEMMNS, paramètre du noyau, 326
SEMMSL, paramètre du noyau, 326
serveurs
de bases de données
configuration, 65
exemples de configuration, 67
matériels, 26, 36
principales versions de Linux, 23
et stations de travail, 18, 19
services UNIX standard, 17
SET SQLPROMPT, commande SQL*Plus, 117
SGA SHOW. *Voir* Oracle Server Manager
SGA, System Global Area, 74
SGBD, site web, 419
SHMMAX, paramètre du noyau, 82, 325
SHMMNI, paramètre du noyau, 325
SHMSEG, paramètre du noyau, 325
SHUTDOWN. *Voir* Oracle Server Manager
SID, identificateur système, 74, 77, 102
Slackware Linux, 21
SMP, matériel multiprocesseur symétrique, 60
SPARC Solaris 2.5.1 et PostgreSQL, 380
SpecTCL, 414
SPIRO-Linux, 24
SQL Pro, site web, 419
SQL*Loader, utilitaire Oracle, 121
SQL*Plus, 116, 119

© copyright Éditions Eyrolles

SQL, compatibilité de base, 130
SQL, site web, 419
SQL3
PostgreSQL et, 364
SQL-92
PostgreSQL et, 364, 420
Progress et, 353
SQLWork, Orasoft, 123
sqsh, client SQL sous licence GNU, 209
ssh, 399
Stampede GNU/Linux, 21, 66
StarOffice, 19, 20
STARTUP. *Voir* Oracle Server Manager
stations de travail, 20
et serveurs, 18, 19
su, commande setuser, 95, 372
support
services Internet, 17
technique. *Voir* assistance
SuSE Linux, 21, 23
svrmgrl, 110
Sybase, 202, 243
accéder au logiciel
ASE 11.9.2, licence payante, 203
SQL Server 11.0.3, licence gratuite, 203
téléchargement, 203
bibliothèques de développement, 209
client
isql, 209
Microsoft SQL Server sous Windows, 214
sqsh, 209
comptes utilisateur, 223, 226
configuration, 210, 226
de la langue, 210
de la machine de destination, 204–206
des bibliothèques client-serveur, 219
des unités de bases de données, 221
du serveur de base de données, 211, 216
du serveur de sauvegarde, 218–219
créer des bases de données, 222
démarrer automatiquement les serveurs, 220
et Java, 210
et les index, 243
et Red Hat Linux, 203
FAQ, 243
format RPM, 203
installer SQL Server 11.0.3, 203, 208
journal de transactions, 244
mot de passe de l'administrateur système, 220
partitions, 206
problèmes, 233, 242
colonne Identité, 233
conformité à SQL, 235
environnement d'exécution, 237
scripts rc, 238
tempdb, 239, 242
utilisation de la mémoire, 237
procédure stockée
sp_addalias, 225
sp_addgroup, 224
sp_addlogin, 223
sp_defaultdb, 223
sp_help, 228
sp_helpdevice, 221
sp_password, 220
réglages, 237
sybinit et Red Hat version 6.1, 210
sybinstall.sh, 207
tester la base de données, 226, 231
sybinstall.sh, script d'installation de Sybase, 207
sybperl, 209
SYS, compte d'administration Oracle, 105
SYSADM, administrateur système DB2, 255, 269
SYSTEM, Oracle
compte d'administration, 105
tablespace, 103
systèmes

© copyright Éditions Eyrolles

d'aide à la décision, 73, 100
hybrides, 73, 100
transactionnels, 73, 100, 318
systèmes de fichiers
à haute performance, 51
Linux natifs, 48

T

tablespaces, 75
emplacement, 103
tar, utilitaire, 292
TCL, 430
Tcl/Tk
interface PostgreSQL, 381, 413
TCP/IP, paramètres, 274, 275
tempdb, Sybase, 239
tempdbs01, Informix, 144
TEMPORARY, tablespaces Oracle, 103
TERM, variable d'environnement
Progress, 329
TNS, Transport Network Substrate, 108
Torvalds, Linus, 10
trap, commande
et Progress, 343
TurboLinux, 21, 22

U

UDBC
pour PostgreSQL, 414
UDF, User-Defined Functions, 257
umask, 84
UNIX
connaissances nécessaires, 69
services standards, 17
utilitaires, 12
unzip, utilitaire, 135
UPS, uninterruptible power supply, 205
Usenet
Informix, 197
USER, Oracle
tablespaces, 103
vue du dictionnaire des données, 119
utilisateurs
et groupes, Sybase, 224
et noms de connexion, Sybase, 223
MySQL, table des, 299
utilitaires UNIX, 12

V

V$, vues des performances dynamiques
Oracle, 120
VA Linux Systems, 26
verrouillage
distribué et Progress, 356
versions commerciales, avantages, 13
VMWare, 16, 36, 69

W

web, exemple d'application. *Voir*
application web
web, plates-formes
d'applications, 428–431
Active Server Pages, 429
Apache, 430
Cold Fusion, 429
Java Server Pages, 429
langages de script, 430
PHP, 430
Zope, 429
WebDB, 121
Windows NT/2000
les avantages de Linux sur, 437
Windows, émulation, 16
WINE, 16
WS-FTP, 18

X

X Window, 13, 16, 85, 86
installation, 60, 62
PostgreSQL et, 372
Progress et, 361
X-Designer, 413
xhost, 85, 372

Y

Yellow Dog Linux, 21

Z

Zend, 413, 432
Zope, 429

© copyright Éditions Eyrolles